

2013

# Iterative Scheme for Solving Optimal Transportation Problems Arising in Reflector Design

Tilman Glimm

*Western Washington University*, [tilmann.glimm@wwu.edu](mailto:tilmann.glimm@wwu.edu)

Nick Henscheid

*University of Arizona*

Follow this and additional works at: [https://cedar.wwu.edu/math\\_facpubs](https://cedar.wwu.edu/math_facpubs)



Part of the [Applied Mathematics Commons](#)

---

## Recommended Citation

Glimm, Tilman and Henscheid, Nick, "Iterative Scheme for Solving Optimal Transportation Problems Arising in Reflector Design" (2013). *Mathematics*. 39.

[https://cedar.wwu.edu/math\\_facpubs/39](https://cedar.wwu.edu/math_facpubs/39)

This Article is brought to you for free and open access by the College of Science and Engineering at Western CEDAR. It has been accepted for inclusion in Mathematics by an authorized administrator of Western CEDAR. For more information, please contact [westerncedar@wwu.edu](mailto:westerncedar@wwu.edu).

## Research Article

# Iterative Scheme for Solving Optimal Transportation Problems Arising in Reflector Design

Tilmann Glimm<sup>1</sup> and Nick Henscheid<sup>2</sup>

<sup>1</sup>Department of Mathematics, Western Washington University, Bellingham, WA 98225, USA

<sup>2</sup>Program in Applied Mathematics, University of Arizona, Tucson, AZ 85716, USA

Correspondence should be addressed to Tilmann Glimm; [glimmt@wwu.edu](mailto:glimmt@wwu.edu)

Received 29 July 2013; Accepted 5 September 2013

Academic Editors: M.-H. Hsu, G. Mishuris, L. Rebollo-Neira, and Q. Song

Copyright © 2013 T. Glimm and N. Henscheid. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We consider the geometric optics problem of finding a system of two reflectors that transform a spherical wavefront into a beam of parallel rays with prescribed intensity distribution. Using techniques from optimal transportation theory, it has been shown previously that this problem is equivalent to an infinite-dimensional linear programming (LP) problem. Here we investigate techniques for constructing the two reflectors numerically by considering the finite-dimensional LP problems which arise as approximations to the infinite-dimensional problem. A straightforward discretization has the disadvantage that the number of constraints increases rapidly with the mesh size, so only very coarse meshes are practical. To address this well-known issue we propose an iterative solution scheme. In each step, an LP problem is solved, where information from the previous iteration step is used to reduce the number of necessary constraints. As an illustration, we apply our proposed scheme to solve a problem with synthetic data, demonstrating that the method allows for much finer meshes than a simple discretization. We also give evidence that the scheme converges. There exists a growing literature for the application of optimal transportation theory to other beam shaping problems, and our proposed scheme is easy to adapt for these problems as well.

## 1. Introduction

The following beam shaping problem from geometric optics was described in [1]; see Figure 1. Suppose a point source emits a spherical wavefront with a given intensity distribution. The problem we are concerned with consists of transforming this input beam into an output beam of parallel light rays with a prescribed intensity distribution. This transformation is to be achieved with a system of two reflectors. The problem has some practical importance in engineering; see further literature cited in [1].

The first rigorous mathematical solution to the problem was provided in [2], using an approach based on the theory of optimal transportation [3–6]. See also the references [7–11] which deal with other beam shaping problems using related techniques.

The result of [2] is summarized in Section 2 below, with Theorem A.5 stating the main result. The central feature is

that the original reflector design problem is reformulated as an infinite-dimensional constrained optimization problem, namely the problem of minimizing a certain linear functional on a function space. It is the dual problem for the problem of finding a map from the input aperture to the output aperture which minimizes a certain cost functional (see Theorem 3 in Section 2 for the exact statement).

This reformulation of the problem not only is of theoretical value for questions of existence and uniqueness of solutions, but it also translates into a practical method for finding the solution. In fact, the discretization of the infinite-dimensional constrained optimization problem is a standard linear programming problem and can be solved numerically. This was already observed for similar beam shaping problems by Glimm and Olikier [7, 8] and independently by Wang [9].

There has been increased interest in numerical methods for optimal transportation problems in the last 10 years. Most work has concentrated on the Monge-Ampère equation,

TABLE 1: Results of runs with the ‘‘Simple Discretization’’ Scheme 5 for several edge lengths  $h$ . Memory was insufficient for the LP solver to handle the case  $h = 0.04$ , so the maximum number of mesh points that could be handled was about 1,750 on each aperture. The data used was the data from Section 4.1. Symbols used:  $h$ : edge length;  $N$ : number of mesh points in  $\bar{D}$ ;  $M$ : number of mesh points in  $\bar{T}$ ; time (s): time in seconds; constr.: number of constraints.

$h$	$M$	$N$	Max error $R_1$	$L_2$ error $R_1$	Max error $R_2$	$L_2$ error $R_2$	Time (s)	Constr.
0.09	511	522	$3.68393E-3$	$1.25443E-3$	$4.58391E-3$	$1.91131E-3$	2.97016	266,742
0.08	660	663	$2.96216E-3$	$7.30738E-4$	$3.39579E-3$	$1.08343E-3$	4.67627	437,580
0.07	880	874	$2.33561E-3$	$7.43916E-4$	$2.83982E-3$	$1.15916E-3$	10.1342	769,120
0.06	1,214	1,214	$1.3051E-3$	$7.12914E-4$	$2.10579E-3$	$8.28341E-4$	17.4949	1,473,796
0.05	1,752	1,757	$9.11319E-4$	$2.66152E-4$	$1.20934E-3$	$4.61313E-4$	45.9136	3,078,264

which arises in the special case of optimal transport in  $\mathbb{R}^n$  with quadratic cost (also called  $L^2$  optimal transport) [12–20], although some authors have treated costs proportional to the distance ( $L^1$  optimal transport) [21]. These methods are based on fluid mechanical approaches [13] or various finite difference approaches [17]. They are generally faster and allow for much larger mesh sizes than methods based on a discretization of the linear programming problem, but since they use the special structure of the Monge-Ampère equation, they are not directly applicable to more general cost functions or more general manifolds.

A linear programming approach similar to the one we investigate here for more general situations for optimal approach was investigated by Rüschemdorf and Uckelmann in 2000 [22] and more recently in two papers by Canavesi et al. [23, 24] who also propose a new algorithm inspired by, but distinct from, a linear programming approach for the single reflector problem treated in [8, 9].

As noted in [22] and other works, an immediate obstacle for the linear programming approach is that the number of constraints becomes very large even for relatively coarse meshes—if there are  $M$  sample points in the input aperture (denoted by  $\bar{D}$  in Figure 1) and  $N$  points in the output aperture ( $\bar{T}$  in Figure 1), then the number of constraints is  $M \cdot N$ . For instance, Rüschemdorf and Uckelmann note in [22] that the LP solver they used, SOPLEX, was designed to handle up to 2 million variables. This corresponds to mesh sizes of approximately 1400 points on the input and output aperture in our problem if one employs a straightforward discretization scheme. We found that, in our implementation, the solver MOSEK (Version 7) could handle about 3 million constraints, corresponding to approximately 1750 sample points on the domains  $\bar{T}$  and  $\bar{D}$  (see Table 1 for details). Rüschemdorf and Uckelmann noted in [22] that, for better results, one needs carefully designed programs. This is what we supply here for our problem: we devised a more elaborate iterative method, where discretized systems are solved on finer and finer grids, in each step, using information from the previous solution to choose only a subset of all possible constraints. This drastically slows the growth of the number of constraints. Details are described in Section 3. With this step-wise mesh refinement scheme, as an illustration, we were able to obtain solutions on meshes with about 10,300 points on each aperture using MATLAB and MOSEK 7, a commercially available LP solver. The number of constraints

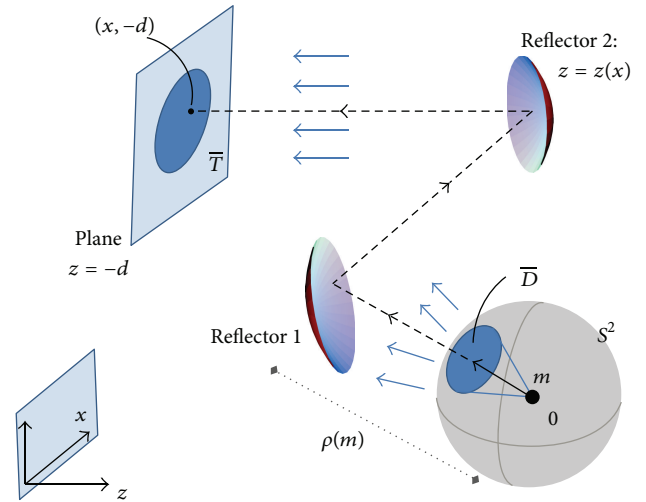


FIGURE 1: Sketch of the reflector problem. The point source is located at the origin  $0$  of the coordinate system. The coordinate system in the lower left-hand corner explains our use of coordinates. The output beam propagates in the direction of the *negative*  $z$ -axis. Points in the plane perpendicular to the  $z$ -axis are denoted by the vector  $\mathbf{x} \in \mathbb{R}^2$ . Correspondingly, points in three-dimensional space are denoted by  $(\mathbf{x}, z)$ . See Section 2 for more. Reprinted from [2].

needed for this was only a fraction of the size of the ‘‘full’’ system. For instance, the run displayed in Table 2 used only 0.33% of all possible constraints.

We note that our proposed algorithm does not make any *a priori* symmetry assumptions on the form the reflectors. It can also be adapted for the numerical solution of other beam shaping problems for which a formulation using optimal transportation theory has been found, for example those in [7, 8, 25].

This article is organized as follows: in Section 2, we recall the reformulation of the reflector construction problem as a linear programming problem as given in [2] and fix some notation. We then describe a basic discretization scheme for the numerical solution and propose an improved ‘‘mesh refinement’’ scheme in Section 3. The following Section 4 is devoted to numerical illustrations and tests of this scheme. We first derive an explicit analytical solution to be used as synthetic data for our numerical work. Then we compute the solution numerically and analyze the error of approximation.

TABLE 2: Results of an iterative run with the “Iterative Refinement” Scheme 6. Memory was insufficient to handle the sixth iteration due to matrix size limitations in MATLAB. Note the run times (last two rows). The bulk of the algorithm run times is taken in setting up the linear programming problem, that is, deciding which constraints to include. Solving the LP once it is set up is fairly fast (last column). The data used was that of Section 4.1, the number of nearest neighbors searched over  $\kappa_{nn} = 6$ , and the quasi-activity threshold was  $\varepsilon = 0.01h^2$ .  $k$ : iteration number;  $M$ : number of mesh points in  $\bar{D}$ ;  $N$ : number of mesh points in  $\bar{T}$ ; constr.: number of constraints; con. dens.: constraint density, that is, used constraints as a percentage of all possible constraints  $M \cdot N$ ; time: computation time for iterative step in seconds, including generating meshes, building, and solving the LP problem; LP time: time in seconds to solve the LP problem.

$k$	$h$	$M$	$N$	Max error $R_1$	$L_2$ error $R_1$	Max error $R_2$	$L_2$ error $R_2$	Constr.	Con. dens.	Time (s)	LP time (s)
0	0.064	1,059	1,059	1.2884E-3	3.77819E-4	1.88783E-3	4.09084E-4	1,121,481	100%	15.962	12.3146
1	0.0512	1,675	1,678	9.40425E-4	3.36918E-4	1.26944E-3	3.63526E-4	56,080	2.00%	208.533	0.938877
2	0.04096	2,647	2,647	7.56437E-4	1.67174E-4	7.53459E-4	1.85922E-4	89,724	1.28%	392.982	1.59831
3	0.032768	4,171	4,175	3.64902E-4	9.63559E-5	5.25941E-4	1.21839E-4	141,550	0.81%	969.239	2.54405
4	0.0262144	6,582	6,567	3.03932E-4	8.8103E-5	5.30343E-4	1.41908E-4	224,327	0.52%	2363.79	4.32524
5	0.02097152	10,313	10,318	2.49854E-4	9.9045E-5	3.9372E-4	1.51851E-4	351,514	0.33%	5909.25	9.21891

We conclude with a brief summary of the advantages and drawbacks of the method and propose future work.

## 2. Formulation as Linear Programming Problem

We now briefly review the notation for the problem posed in [1], as well as the result of [2], which reformulates the reflector design problem as a linear programming problem. More details are contained in Appendix A.

Consider the configuration shown in Figure 1. A point source at the origin  $O = (0, 0, 0)$  generates a spherical wavefront over a given input aperture  $\bar{D}$  contained in the unit sphere  $S^2$ . We use the bar in the notation for  $\bar{D}$  to stress that  $\bar{D}$  is a closed set. The input beam has a given intensity distribution. By means of two reflectors, this wavefront is to be transformed into a beam of parallel rays propagating in the direction of the negative  $z$ -axis. This output beam is required to have a prescribed intensity distribution. The cross section of the output beam in a plane perpendicular to the direction of propagation is called the output aperture and is denoted by  $\bar{T}$ . Again, we use the notation  $\bar{T}$  to stress that  $\bar{T}$  is closed. Certain regularity conditions apply to  $\bar{D}$  and  $\bar{T}$ ; see [2] for the technical details.

Denote points in space  $\mathbb{R}^3$  by pairs  $(\mathbf{x}, z)$ , where  $\mathbf{x} \in \mathbb{R}^2$  is the position vector in a plane perpendicular to the direction of propagation and  $z \in \mathbb{R}$  is the coordinate in the (negative) direction of propagation. See again Figure 1 for our convention on the direction of the  $z$ -axis. Points on the unit sphere  $S^2$  will typically be denoted by  $\mathbf{m} \in S^2$ ; their components are also written as  $\mathbf{m} = (\mathbf{m}_x, m_z)$  with  $|\mathbf{m}_x|^2 + m_z^2 = 1$ .

We fix the output aperture in the plane  $z = -d$ . We will seek to represent the first reflector as the graph of its polar radius  $\rho(\mathbf{m})$  (for  $\mathbf{m} \in \bar{D}$ ) and the second reflector as the graph of a function  $z(\mathbf{x})$  (for  $z \in \bar{T}$ ). See Figure 1. That is

$$\begin{aligned} \text{Reflector 1: } R_1 &= \{ \rho(\mathbf{m}) \cdot \mathbf{m} \mid \mathbf{m} \in \bar{D} \}, \\ \text{Reflector 2: } R_2 &= \{ (\mathbf{x}, z(\mathbf{x})) \mid \mathbf{x} \in \bar{T} \}. \end{aligned} \tag{1}$$

The geometrical optics approximation is assumed. It follows from general principles of geometric optics that all rays will have equal length from  $(0, 0, 0)$  to the plane  $z = -d$ ; this length is called the optical path length and will be denoted by  $L$ . We define the *reduced optical path length* as  $\ell = L - d$ .

Oliker [1] showed that local energy conservation translates into a complicated partial differential equation of Monge-Ampère type for  $\rho(\mathbf{m})$ . As noted in [1], the resulting equation is quite involved, and a rigorous analysis of this equation seems very difficult. See equation (59) in [1].

To amend this, the problem was reformulated in [2] as an infinite dimensional linear programming problem, which makes a complete analysis possible, both concerning theoretical results on existence and uniqueness, and gives a method for practical computations.

For this, the following function  $K(\mathbf{m}, \mathbf{x})$ , called the cost function in analogy with the theory of optimal transportation, plays an important role:

$$\begin{aligned} K(\mathbf{m}, \mathbf{x}) &= \frac{\ell - \langle \mathbf{m}_x, \mathbf{x} \rangle}{2\ell(\ell^2 - |\mathbf{x}|^2)(1 + m_z)} - \frac{1}{4\ell^2} \\ \text{for } \mathbf{m} &= (\mathbf{m}_x, m_z) \in \bar{D}, \mathbf{x} \in \bar{T}. \end{aligned} \tag{2}$$

In further preparation, the following two transformations are needed.

*Definition 1.* Let  $z = z(\mathbf{x})$  be a continuous function defined on  $\bar{T} \subseteq \mathbb{R}^2$ . Then define the function

$$\tilde{z}(\mathbf{x}) = \frac{1}{2\ell} - \frac{z(\mathbf{x})}{\ell^2 - |\mathbf{x}|^2} \quad \text{for } \mathbf{x} \in \bar{T}. \tag{3}$$

*Definition 2.* Let  $\rho = \rho(\mathbf{m})$  be a continuous function defined on  $\bar{D} \subseteq S^2$  with  $\rho > 0$ . Then define the function

$$\hat{\rho}(\mathbf{m}) = -\frac{1}{2\ell} + \frac{1}{2\rho(\mathbf{m}) \cdot (m_z + 1)} \quad \text{for } \mathbf{m} \in \bar{D}. \tag{4}$$

The following characterization of the solution  $(\rho, z)$  of the reflector problem described above was given in [2].

**Theorem 3.** *If the pair  $(\rho, z)$  is a solution of the above reflector design problem, then the transformed pair  $(\log \hat{\rho}, \log \bar{z})$  minimizes the functional*

$$\mathcal{F}(r, \zeta) = \int_{\bar{D}} r(\mathbf{m}) I(\mathbf{m}) d\sigma + \int_{\bar{T}} \zeta(\mathbf{x}) L(\mathbf{x}) dx \quad (5)$$

on the function space  $\text{Adm}(\bar{D}, \bar{T}) = \{(r, \zeta) \in C(\bar{D}) \times C(\bar{T}) \mid r(\mathbf{m}) + \zeta(\mathbf{x}) \geq \log K(\mathbf{m}, \mathbf{x}) \text{ for all } \mathbf{m} \in \bar{D}, \mathbf{x} \in \bar{T}\}$ .

One can solve the reflector construction problem by solving the infinite-dimensional LP problem in the above theorem and then applying the inverse transformations of (3) and (4) to recover  $\rho(\mathbf{m})$  and  $z(\mathbf{x})$ . In the remainder of this paper, we will concentrate on numerical solutions for this problem.

It is also possible to characterize the ray-tracing map analytically.

**Definition 4.** Let  $(\rho, z) \in C(\bar{D}) \times C(\bar{T})$  be a solution of the reflector problem. Define its *ray-tracing map* as a set-valued map  $\gamma : \bar{D} \rightarrow \{\text{subsets of } \bar{T}\}$  via

$$\gamma(\mathbf{m}) = \{\mathbf{x} \in \bar{T} \mid \hat{\rho}(\mathbf{m}) \bar{z}(\mathbf{x}) = K(\mathbf{m}, \mathbf{x})\} \quad \text{for } \mathbf{m} \in \bar{D}. \quad (6)$$

In [2], it is shown that  $\gamma(\mathbf{m})$  is in fact single-valued for almost all  $\mathbf{m} \in \bar{D}$ . Therefore,  $\gamma$  may be regarded as a transformation  $\gamma : \bar{D} \rightarrow \bar{T}$ . Indeed, one can show that a ray emitted from the origin in the direction  $\mathbf{m} \in \bar{D}$  will be reflected to a ray in the negative  $z$ -direction labeled by  $\mathbf{x} = \gamma(\mathbf{m}) \in \bar{T}$ ; see [2].

More details and rigorous statements are found in Appendix A.

### 3. Numerical Schemes

In the following, we describe two schemes for solving the minimization problem in Theorem 3 numerically. First, we list the given data on which the solution depends.

- (i) Domains  $\bar{D} \subseteq S^2$ ,  $\bar{T} \subseteq \mathbb{R}^2$  (The rigorous result in [2] required the additional technical assumption  $(0, 0, -1), (0, 0, 1) \notin \bar{D}$ . For practical purposes, these assumptions can often be dropped.)
- (ii) Nonnegative integrable functions  $I(\mathbf{m})$ ,  $\mathbf{m} \in \bar{D}$ , and  $L(\mathbf{x})$ ,  $\mathbf{x} \in \bar{T}$ , with  $\int_{\bar{D}} I d\sigma = \int_{\bar{T}} L dx$ .
- (iii) A reduced optical length  $\ell$ .
- (iv) The value  $\rho(\mathbf{m}_1) = \rho_1$  for some fixed  $\mathbf{m}_1 \in \bar{D}$ , where  $\rho$  is the radial function of the first reflector. (Note that without this constraint, the reflectors are not uniquely determined. This can be seen in the expression for the functional in Theorem 3. Indeed, any transformation of the form  $r \mapsto r + c$ ,  $\zeta \mapsto z - c$  for constant  $c$  will leave the objective functional unchanged.)

The first numerical scheme, Scheme 5 is a straightforward discretization of the minimization problem in Theorem 3 via meshes on the domains  $\bar{D}$  and  $\bar{T}$ .

As described in the introduction, this scheme is memory-intensive because every pair of points from  $\bar{D}$  and  $\bar{T}$  gives rise to a constraint. To address this issue, we propose a more sophisticated scheme, Scheme 6, which is an iterative scheme using finer and finer meshes, where, in each iteration, information from the previous solution is utilized to reduce the number of constraints.

**3.1. "Simple Discretization" Scheme.** The characterization of solutions to the reflector problem by means of the minimization of a linear functional in Theorem 3 immediately suggests a solution algorithm by means of a discretization. Essentially, the same algorithm was used for different problems in [22–24].

**Scheme 5 (simple discretization).** (i) Create a mesh in the input domain  $\bar{D}$  by choosing sets  $d_1^{(0)}, d_2^{(0)}, \dots, d_{M^{(0)}}^{(0)} \subseteq \bar{D}$ , where the interiors of any  $d_i^{(0)}$  and  $d_j^{(0)}$  are disjoint (for  $i \neq j$ ) and  $\bar{D} = \bigcup_{i=1}^{M^{(0)}} d_i^{(0)}$ . For instance,  $d_1^{(0)}, d_2^{(0)}, \dots, d_{M^{(0)}}^{(0)}$  may be a triangulation of  $\bar{D}$ .

(ii) Similarly, create a mesh in the output domain  $\bar{T}$  by choosing sets  $t_1^{(0)}, t_2^{(0)}, \dots, t_{N^{(0)}}^{(0)} \subseteq \bar{T}$ , where the interiors of any  $t_i^{(0)}$  and  $t_j^{(0)}$  are disjoint (for  $i \neq j$ ) and  $\bar{T} = \bigcup_{j=1}^{N^{(0)}} t_j^{(0)}$ .

(iii) Choose sample points  $\mathbf{m}_i^{(0)} \in d_i^{(0)}$  (for  $i = 1, \dots, M^{(0)}$ ) and  $\mathbf{x}_j^{(0)} \in t_j^{(0)}$  (for  $j = 1, \dots, N^{(0)}$ ). Here, we may assume that  $\mathbf{m}_1^{(0)} = \mathbf{m}_1$  is the same point as given in the data in (iv) above.

(iv) Find the solution  $(\{r_i^{(0)}\}_{i=1}^{M^{(0)}}, \{\zeta_j^{(0)}\}_{j=1}^{N^{(0)}})$  of the following LP problem:

$$\text{Minimize } \sum_{i=1}^{M^{(0)}} r_i^{(0)} I(\mathbf{m}_i^{(0)}) \text{ area}(d_i^{(0)}) \quad (7)$$

$$+ \sum_{j=1}^{N^{(0)}} \zeta_j^{(0)} L(\mathbf{x}_j^{(0)}) \text{ area}(t_j^{(0)})$$

$$\text{subject to } r_1^{(0)} = \log \hat{\rho}_1, \quad (8)$$

where  $\hat{\rho}_1$  is as given in the data in (iv) above and

$$r_i^{(0)} + \zeta_j^{(0)} \geq \log K(\mathbf{m}_i^{(0)}, \mathbf{x}_j^{(0)}) \quad (9)$$

for  $i = 1, \dots, M^{(0)}$ ,  $j = 1, \dots, N^{(0)}$ .

- (v) Find the numbers  $\rho_i^{(0)}$ ,  $i = 1, \dots, M^{(0)}$  and  $z_j^{(0)}$ ,  $j = 1, \dots, N^{(0)}$  such that  $r_i^{(0)} = \log \hat{\rho}_i^{(0)}$  and  $\zeta_j^{(0)} = \log \bar{z}_j^{(0)}$ . This is straightforward by taking the inverse of the transformations given in Definition 1 and Scheme 5.

Then  $\rho_i^{(0)}$  is an approximation for the true value  $\rho(\mathbf{m}_i^{(0)})$  of the radial function of the first reflector, evaluated at the sample point  $\mathbf{m}_i^{(0)}$  for  $i = 1, \dots, M^{(0)}$ . Similarly,  $z_j^{(0)}$  is an approximation for the true value  $z(\mathbf{x}_j^{(0)})$  of the function describing the second reflector, evaluated at the sample point



$\mathbf{x}_j^{(0)}$  for  $j = 1, \dots, N^{(0)}$ . We are using the superscript “(0)” here to distinguish this solution from additional iterative approximations we will obtain with the iterative scheme described below.

Note that (7) is a discretization of the sum of integrals in the functional  $\mathcal{F}(r, \zeta) = \int_{\bar{D}} r(\mathbf{m})I(\mathbf{m})d\sigma + \int_{\bar{T}} \zeta(\mathbf{x})L(\mathbf{x})dx$ . We have chosen the simplest discretization here. Depending on the geometry of the problem, other discretization schemes [26] may yield better results. In practice, we have chosen partitions  $d_i$  and  $t_j$  of approximately uniform size and diameter, but depending on the form of the intensity distributions, other partitions may be more appropriate.

It is important to point out that this discretization also yields a discretized version of the ray-tracing map  $\gamma$  (see Definition 4). Namely, for each index  $i = 1, \dots, M^{(0)}$ , there is at least one corresponding index  $j^*$ ,  $1 \leq j^* \leq N^{(0)}$  where the constraint corresponding to the pair  $(i, j^*)$  is active; that is, where we have

$$r_i^{(0)} + \zeta_{j^*}^{(0)} = \log K(\mathbf{m}_i^{(0)}, \mathbf{x}_{j^*}^{(0)}). \quad (10)$$

This means that the point  $\mathbf{m}_i^{(0)}$  is approximately the image of the point  $\mathbf{x}_{j^*}^{(0)}$  under the ray-tracing map  $\gamma$ .

**3.2. “Iterative Refinement” Scheme.** As noted in the introduction, one of the drawbacks of Scheme 5 is that the constraint set in the linear programming problem in the penultimate step becomes large very fast with finer meshes, and the corresponding problem becomes too large to handle for standard LP solvers. We were not able to solve problems with more than about 1750 mesh points on each aperture on a standard PC with 4 GB RAM ( $M^{(0)} \approx 1750$ ,  $N^{(0)} \approx 1750$ ).

We addressed this problem by developing an iterative scheme. First, the problem is solved for a mesh with relatively few sample points (say  $M^{(0)} = N^{(0)} = 1000$ ). Then, a finer mesh is chosen, and the previous solution is used to reduce the number of constraints.

Specifically, we have the following scheme, depending on a number  $\varepsilon^{(1)} > 0$ , which we call the “quasi-activity threshold,” and an integer  $\kappa_{\text{nn}}$ , the nearest neighbor search depth, to be explained in detail below.

**Scheme 6** (iterative refinement). Use Scheme 5 to find an initial solution  $(\{r_i^{(0)}\}_{i=1}^{M^{(0)}}, \{\zeta_j^{(0)}\}_{j=1}^{N^{(0)}})$  for  $M^{(0)}$  sample points on  $\bar{D}$  and  $N^{(0)}$  sample points on  $\bar{T}$ , respectively.

**Definition 7.** Call a pair  $(i, j)$  with  $i \in \{1, \dots, M^{(0)}\}$  and  $j \in \{1, \dots, N^{(0)}\}$  “quasi-active” if

$$r(\mathbf{m}_i^{(0)}) + \zeta(\mathbf{x}_j^{(0)}) - \log K(\mathbf{m}_i^{(0)}, \mathbf{x}_j^{(0)}) \leq \varepsilon^{(0)}. \quad (11)$$

Note that the pair is “active” in the standard sense of linear programming theory if the above term is equal to zero.

- (i) Create a second, finer mesh on  $\bar{D}$  by choosing sets  $d_1^{(1)}, d_2^{(1)}, \dots, d_{M^{(1)}}^{(1)} \subseteq \bar{D}$ , where the interiors of any  $d_i^{(1)}$  and  $d_j^{(1)}$  are disjoint (for  $i \neq j$ ) and  $\bar{D} = \bigcup_{i=1}^{M^{(1)}} d_i^{(1)}$ .

Here,  $M^{(1)}$  is chosen larger than  $M^{(0)}$ , meaning that we have a finer mesh.

- (ii) Similarly, create a second mesh on  $\bar{T}$  by choosing sets  $t_1^{(1)}, t_2^{(1)}, \dots, t_{N^{(1)}}^{(1)} \subseteq \bar{T}$ , where the interiors of any  $t_i^{(1)}$  and  $t_j^{(1)}$  are disjoint (for  $i \neq j$ ) and  $\bar{T} = \bigcup_{j=1}^{N^{(1)}} t_j^{(1)}$ .
- (iii) Choose sample points  $\mathbf{m}_i^{(1)} \in d_i^{(1)}$  (for  $i = 1, \dots, M^{(1)}$ ) and  $\mathbf{x}_j^{(1)} \in t_j^{(1)}$  (for  $j = 1, \dots, N^{(1)}$ ). Here, we may assume that  $\mathbf{m}_1^{(1)} = \mathbf{m}_1$  is the same point as given in the data in (iv) above.
- (iv) For each pair  $(i, j)$  with  $i = 1, \dots, M^{(1)}$ , and  $j = 1, \dots, N^{(1)}$ , find the  $\kappa_{\text{nn}}$  nearest neighbors of  $\mathbf{m}_i^{(1)}$  from the previous mesh on  $\bar{D}$  and the  $\kappa_{\text{nn}}$  nearest neighbors of  $\mathbf{x}_j^{(1)}$  from the previous mesh on  $\bar{T}$ .

**Definition 8.** Say that the pair  $(i, j)$  is *potentially active* if at least one of the  $\kappa_{\text{nn}}^2$  pairs consisting of a nearest neighbor of  $\mathbf{m}_i^{(1)}$  and a nearest neighbor of  $\mathbf{x}_j^{(1)}$  is quasi-active in the above sense.

- (i) Find the solution  $(\{r_i\}_{i=1}^{M^{(1)}}, \{\zeta_j^{(1)}\}_{j=1}^{N^{(1)}})$  of the following LP problem:

$$\begin{aligned} \text{Minimize} \quad & \sum_{i=1}^{M^{(1)}} r_i^{(1)} I(\mathbf{m}_i^{(1)}) \text{area}(d_i^{(1)}) \\ & + \sum_{j=1}^{N^{(1)}} \zeta_j^{(1)} L(\mathbf{x}_j^{(1)}) \text{area}(t_j^{(1)}) \end{aligned} \quad (12)$$

$$\text{subject to} \quad r_1^{(1)} = \log \hat{\rho}_1,$$

$$r_i^{(1)} + \zeta_j^{(1)} \geq \log K(\mathbf{m}_i^{(1)}, \mathbf{x}_j^{(1)})$$

for any potentially active pair

(in the above sense).

- (ii) Find the numbers  $\rho_i^{(1)}$ ,  $i = 1, \dots, M^{(1)}$  and  $z_j^{(1)}$ ,  $j = 1, \dots, N^{(1)}$  such that  $r_i^{(1)} = \log \hat{\rho}_i^{(1)}$  and  $\zeta_j^{(1)} = \log \hat{z}_j^{(1)}$ .

The idea of Scheme 6 is to solve the discretized LP on a coarse mesh first and then use this information to reduce the number of constraints needed for the LP on a finer mesh. A key difference between Schemes 5 and 6 is that in the discretized LP problem, in Scheme 6, not all pairs of sample points from  $\bar{D}$  and  $\bar{T}$ , represented by pairs of indices  $(i, j)$ , are included in the list of constraints. In fact, we would in principle only need to include those where the constraint is active, that is, where the corresponding inequality holds with equality. Of course, this information is not available *a priori*. Instead, we use the following heuristic. A constraint should only be included in the LP problem if the corresponding point on the second reflector is close to the image of the corresponding point on the first reflector under the ray-tracing map. In terms of the LP problem formulation, the

corresponding pair of points is located close to an active pair from the previous iteration. We call such a constraint “potentially active.” Our heuristic is that a constraint should be considered potentially active if there is a pair of its respective  $\kappa_{\text{nn}}$  nearest neighbors which corresponds to a quasi-active constraint from the previous iteration. Here, “quasi-active” constraints include active constraints (where the constraint is attained, i.e., the term  $r(\mathbf{m}_j^{(0)}) + \zeta(\mathbf{x}_j^{(0)}) - \log K(\mathbf{m}_i^{(0)}, \mathbf{x}_j^{(0)})$  is zero) and those which are “almost active” in the sense that the term  $r(\mathbf{m}_j^{(0)}) + \zeta(\mathbf{x}_j^{(0)}) - \log K(\mathbf{m}_i^{(0)}, \mathbf{x}_j^{(0)})$  is positive, but small. “Smallness” is encoded in the quasi-activity threshold  $\varepsilon^{(1)}$ . Clearly, increasing  $\varepsilon^{(1)}$  means more constraints are included, but also potentially the solution of the LP represents a better approximation of the exact reflector pair.

In our numerical tests, we found it advantageous to scale the functions  $I(\mathbf{m})$  and  $L(\mathbf{x})$  so that the approximations for the integrals  $\int_{\bar{D}} I d\sigma$  and  $\int_{\bar{T}} L dx$  yield exactly the same result.

Note that the algorithm in Scheme 6 can be iterated again. We can use the first iterative solution  $\rho_i^{(1)}$ ,  $i = 1, \dots, M^{(1)}$  and  $z_j^{(1)}$ ,  $j = 1, \dots, N^{(1)}$ , in Scheme 6, to obtain a second iterative solution  $\rho_i^{(2)}$ ,  $i = 1, \dots, M^{(2)}$  and  $z_j^{(2)}$ ,  $j = 1, \dots, N^{(2)}$ , and so on.

Note that (7) is a discretization of the sum of integrals in the functional  $\mathcal{F}(r, \zeta) = \int_{\bar{D}} r(\mathbf{m}) I(\mathbf{m}) d\sigma + \int_{\bar{T}} \zeta(\mathbf{x}) L(\mathbf{x}) dx$ . We note that we have chosen again the simplest discretization here for the sum of integrals  $\int_{\bar{D}} r(\mathbf{m}) I(\mathbf{m}) d\sigma + \int_{\bar{T}} \zeta(\mathbf{x}) L(\mathbf{x}) dx$ . As in Scheme 6, depending on the geometry of the problem, other discretization schemes [26] may yield better results.

**3.3. Choice of Thresholds for Iterative Refinement Scheme 6.** For the iteration of Scheme 6 described above, we have to choose a corresponding sequence of mesh sizes  $\{(M^{(0)}, N^{(0)}), (M^{(1)}, N^{(1)}), (M^{(2)}, N^{(2)}), \dots\}$ . We also need to choose a corresponding sequence of thresholds  $\varepsilon^{(1)}, \varepsilon^{(2)}, \varepsilon^{(3)}, \dots$ .

This last question is of a certain practical importance. Indeed, if the thresholds  $\{\varepsilon^{(k)}\}_k$  are “high” (e.g., if they decrease slowly), then “many” constraints are included in each iteration, meaning that the size of the LP problems increases quickly. If the sequence  $\{\varepsilon^{(k)}\}_k$  is “low”, then “few” constraints are included in each iteration. This is of course in principle desirable, but if the thresholds are too low, this could cause that some index  $i$  (with  $1 \leq i \leq M^{(k)}$ ) may not even be included in any of the constraints with any of the indices  $j$  (with  $1 \leq j \leq N^{(k)}$ ) or vice versa. This would cause the LP problem to be unbounded, and hence, there would be no solution.

We can make a very rough estimate for a good choice of the sequence  $\{\varepsilon^{(k)}\}_k$ , assuming for simplicity that  $M^{(k)} \approx N^{(k)}$ . Let us assume that each sample point  $m_i^{(k)}$  in the input aperture  $\bar{D}$  is paired via the ray-tracing map  $\gamma$  with a unique point  $\mathbf{x}_{j(i)}^{(k)} \approx \gamma(m_i^{(k)})$  and vice versa. Thus, there are  $M^{(k)}$  such pairs. Let  $r(\mathbf{m}), \mathbf{m} \in \bar{D}$  and  $\zeta(\mathbf{x}), \mathbf{x} \in \bar{T}$  denote the

solutions of the reflector problem. Then the set of all points in  $(\mathbf{m}, \mathbf{x})$ -space close to a given pair  $(\mathbf{m}_i^{(k)}, \mathbf{x}_{j(i)}^{(k)})$  that satisfy  $r(\mathbf{x}) - z(\mathbf{x}) - \log K(\mathbf{m}, \mathbf{x}) < \varepsilon^{(k)}$  is approximately an ellipsoid for small  $\varepsilon^{(k)}$ . This can be seen by using the Taylor expansion around the pair  $(\mathbf{m}_i^{(k)}, \mathbf{x}_{j(i)}^{(k)})$  in  $(\mathbf{m}, \mathbf{x})$ -space. The volume of this ellipsoid is proportional to  $(\varepsilon^{(k)})^2$ . So the proportion of pairs of points that satisfy the above constraint among all constraints is roughly proportional to  $(\varepsilon^{(k)})^2$ . Since each pair  $(\mathbf{m}_i^{(k)}, \mathbf{x}_j^{(k)})$  that satisfies this inequality corresponds to a constraint included in the LP of the  $k$ th iteration, the total number of constraints should be roughly proportional to  $(M^{(k)})^3 \cdot (\varepsilon^{(k)})^2$ . So to keep the number of constraints growing linearly throughout the iterations, based on these heuristics, one may choose

$$\varepsilon^{(k)} = \frac{C}{M^{(k)}}, \quad (13)$$

where  $C$  is a constant. These heuristics are very rough. In practice, we have found that when using the formula  $\varepsilon^{(k)} = C(M^{(k)})^\alpha$ , there is a range of values of  $\alpha$  which all produce similar results. See Section 4.2.

## 4. Numerical Tests

To test the validity of the numerical scheme described above, we used it on a case where the solution is known in analytic form. In the next subsection, we first describe this special analytic solution and then discuss our results.

**4.1. A Data Set.** We have constructed a particular data set using an explicit analytical solution described in more detail in Appendix B. The data set of this section is obtained from the more general solution of Appendix B by choosing  $a = b = 0$  and  $c = -0.4$ ,  $\alpha = 1$ , and  $R = 1.3$ . We consider the input aperture

$$\bar{\Omega} = \left\{ (m_x, m_y, m_z) \in S^2 \mid \sqrt{m_x^2 + m_y^2} \leq 0.8, m_z < 0 \right\}, \quad (14)$$

which is a spherical cap centered at the point  $(0, 0, -1)$ . The data do not represent a physically possible set of reflectors since there would be blockage. However, the example is useful as an illustration, and the numerical scheme itself is completely independent of the shape of the apertures or any *a priori* symmetries of the problem. Using the results from Appendix B, we can now find the output aperture in a straightforward manner as follows:

$$\bar{T} = \left\{ (x, y) \in \mathbb{R}^2 \mid \sqrt{x^2 + y^2} \leq \frac{17}{9} \right\}. \quad (15)$$

We choose a constant output intensity

$$L(\mathbf{x}) = 1 \quad \text{for } \mathbf{x} \in \bar{T}. \quad (16)$$

Using the relation (B.7), this gives the input intensity

$$I(\mathbf{m}) = \frac{14.2716049383}{(1 - m_z)^2} \quad \text{for } \mathbf{m} = (m_x, m_y, m_z) \in \bar{\Omega}. \quad (17)$$

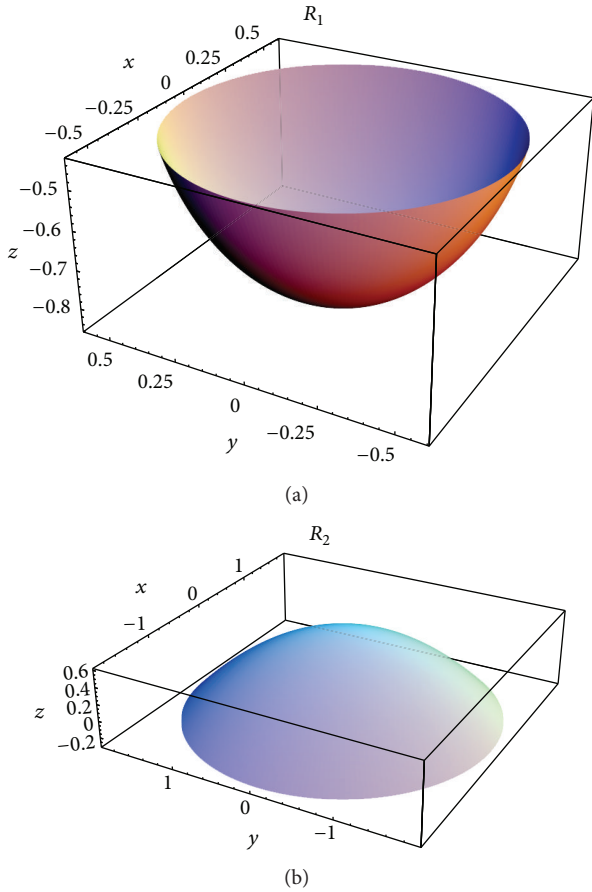


FIGURE 2: Plots of the reflectors given by (18) (a) and (19) (b) in the explicit data set given in Section 4.1.

The two reflectors are given as follows:

$$\text{radial radius } \rho \text{ for } R_1: \rho(\mathbf{m}) = \frac{0.765}{1.3 + 0.4 m_z}, \quad (18)$$

$$\text{equation for } R_2: z = -0.25(x^2 + y^2) + 0.6. \quad (19)$$

The corresponding reduced optical path length is

$$\ell = R + 2\alpha = 2.9. \quad (20)$$

The two reflectors are shown in Figure 2.  $R_1$  is part of a spheroid, and  $R_2$  is part of a paraboloid; see Appendix B for details.

4.2. Results. Using the data set from Section 4.1, we conducted a series of numerical tests to establish the validity of the numerical Scheme 6. Since there are explicit analytical expressions for the two solutions  $\rho(m)$  and  $z(x)$  describing the two reflectors as given in (18) and (19), respectively, we were able to compute the error of approximation directly. See also Figure 2 for plots of these reflectors.

The Schemes 5 and 6 were implemented in MATLAB with the package distmesh [27] for mesh generation in each iteration step and the commercial linear programming

solver MOSEK (version 7) [28] for solving the resulting LP problems. Computations were performed on an Intel Core i7-2620M/2.70 GHz with 4 GB RAM, running Windows 7 (64 bit).

The mesh generation algorithm is based on a relaxation scheme of forces in a truss structure [27]. For this, the input is the desired edge length  $h$ , not the number of mesh points  $M$  (for the domain  $\bar{\Omega}$ ) or  $N$  (for the domain  $\bar{T}$ ), respectively. However, it can be seen from geometric considerations that the relation between the average edge length  $h$  and the number of mesh points  $M$  on  $\bar{D}$  obeys

$$h \sim \frac{1}{\sqrt{M}}. \quad (21)$$

In the tables below, we show both the number of mesh points  $M, N$  and the edge length  $h$ , as we believe both are informative.

Our primary aim is to demonstrate the practicability of the proposed Scheme 6. For comparison, we first tested the performance of the ‘‘Simple Discretization’’ Scheme 5. The results are summarized in Table 1. The maximum mesh sizes attainable are about 1750 points, corresponding to approximately 3 million constraints. We found that this was the maximum size for the LP solver.

To test Scheme 6, we chose a sequence of desired edge lengths  $h^{(0)}, h^{(1)}, \dots$  by reducing  $h$  by 20% in each step; that is,  $h^{(k)} = 0.8^k h^{(0)}$ . Then, based on the heuristics given above and in Section 3.3, we determined the corresponding constraint thresholds with the following formula:

$$\varepsilon^{(k)} = C \cdot (h^{(k)})^\mu, \quad (k = 1, 2, 3, \dots), \quad (22)$$

where  $C$  and  $\mu$  are constants. The heuristic arguments in Section 3.3 yield  $\mu = 2$ , but we also tested other values. As indicated in Section 3.3, ‘‘large’’ values for  $C$  mean that ‘‘many’’ constraints are included in the LP in each step, which potentially means good approximations, but also high memory usage. In contrast, ‘‘small’’ values for  $C$  improve memory usage but may mean that the LP problem may become unbounded. This is the case if constraints that would be active in the LP problem comprising of all constraints are left out in an iterated LP problem. In our tests, the only time this happened was when we chose  $C = 0$ , that is, if the quasi-activity threshold is set to zero and only active constraints are considered ‘‘quasi-active.’’ We conclude that our results are very robust with respect to  $C$  and  $\mu$ , meaning for a broad range of values, we reached the terminal iteration with  $h = 0.02097152$ ; see Table 2. This is because the memory limiting issue in our tests was not the LP solver but the maximum size of arrays in MATLAB when computing the cost matrix; see paragraph below. We also conducted some informal tests of the algorithm dependence on the output and input intensities  $I(\mathbf{m})$  and  $L(\mathbf{x})$ , specifically to know whether the algorithm is stable under small perturbations of  $I(\mathbf{m})$  and  $L(\mathbf{x})$ . For this, we added small perturbation  $\delta \cos(m_y)$  to  $I(\mathbf{m})$  and  $\delta \cos(x)$  to  $L(x)$ , then renormalized  $L(\mathbf{x})$  so that the two integrals were the same. We found that the algorithm was indeed stable in the sense that the same maximum number of points was reached for different values of  $\delta$ .



TABLE 3: Results of the final iteration of the “Iterative Refinement” Scheme 6 with different values of the nearest neighbor search depth  $\kappa_{nn}$ . All other data used was the same as in Table 2. Note that the nearest neighbor search depth  $\kappa_{nn}$  has a large influence on the size of the LP problem but relatively little influence on the errors or run times. Column labels are identical to Table 2 except for the first column.

$\kappa_{nn}$	$h$	$M$	$N$	Max error $R_1$	$L_2$ error $R_1$	Max error $R_2$	$L_2$ error $R_2$	Constr.	Con. dens.	Time (s)	LP time (s)
2	0.02097152	10,313	10,318	$2.71973E-4$	$9.04017E-5$	$4.38374E-4$	$1.37601E-4$	96256	0.090%	5947.76	2.82425
4	0.02097152	10,313	10,318	$2.48492E-4$	$9.77635E-5$	$3.92226E-4$	$1.50005E-4$	226006	0.212%	5752.17	6.09849
6	0.02097152	10,313	10,318	$2.49854E-4$	$9.9045E-5$	$3.9372E-4$	$1.51851E-4$	351514	0.330%	5909.25	9.21891
8	0.02097152	10,313	10,318	$2.49816E-4$	$9.90729E-5$	$3.94393E-4$	$1.5188E-4$	482058	0.453%	6322.4	12.9115

Results of a typical run are summarized in Table 2. Our results show that the proposed scheme is easy to implement and makes it possible to use much finer meshes than a simple discretization scheme. The largest number of points were about 10,000 on each aperture. It is notable that what prevented our tests to include more plot points was not the LP solver but memory problems related to the maximum size of matrices in MATLAB during the set-up phase of the LP problem when computing the matrix of cost coefficients  $K_{ij}$ , which is of size  $M^{(k)} \times N^{(k)}$ . For the configuration we used, the maximum size of arrays in MATLAB is about 280 million variables, see [29].

These results give indication that the scheme converges and that the maximum error for the first reflector decreases proportional to  $M^\sigma$ , or equivalently proportional to  $h^{2\sigma}$ , where  $\sigma = -0.773$ , data from Table 2. The same formulation for scheme 5 yields an exponent of  $\sigma = -1.181$ , unsurprisingly a slightly more rapid decrease, data from Table 1. The geometric distribution of errors is indicated in Figure 3.

To investigate the role of the nearest neighbor search depth  $\kappa_{nn}$ , we performed several tests with different values. The results are summarized in Table 3. The value of  $\kappa_{nn}$  has very little influence on the error of approximation. This makes intuitive sense as the crucial issue in each iteration is that those constraints which are active in the “full” LP problem (i.e., the one that includes all possible constraints) are included. As long as all active constraints are included, one expects to get the same solution. Note, however, that the number of constraints changes quite drastically with  $\kappa_{nn}$ , ranging from 96,256 (0.09% of all possible constraints) for  $\kappa_{nn} = 2$  to 48,2058 (0.45% of all possible constraints) for  $\kappa_{nn} = 8$  in this example. The time to solve the LP problems, thus, also varies from 2.82 seconds for  $\kappa_{nn} = 2$  to 12.91 seconds for  $\kappa_{nn} = 8$ , although those gains are dwarfed by the overall run times of the algorithm which include assembling the LP problem.

## 5. Conclusion and Future Work

We investigated two numerical schemes for solving an infinite-dimensional optimal transportation problem arising in reflector design, a straightforward discretization and a novel iterative scheme, which uses knowledge of the previous solution in each step to reduce the number of constraints. The latter scheme is easily adapted to similar transportation

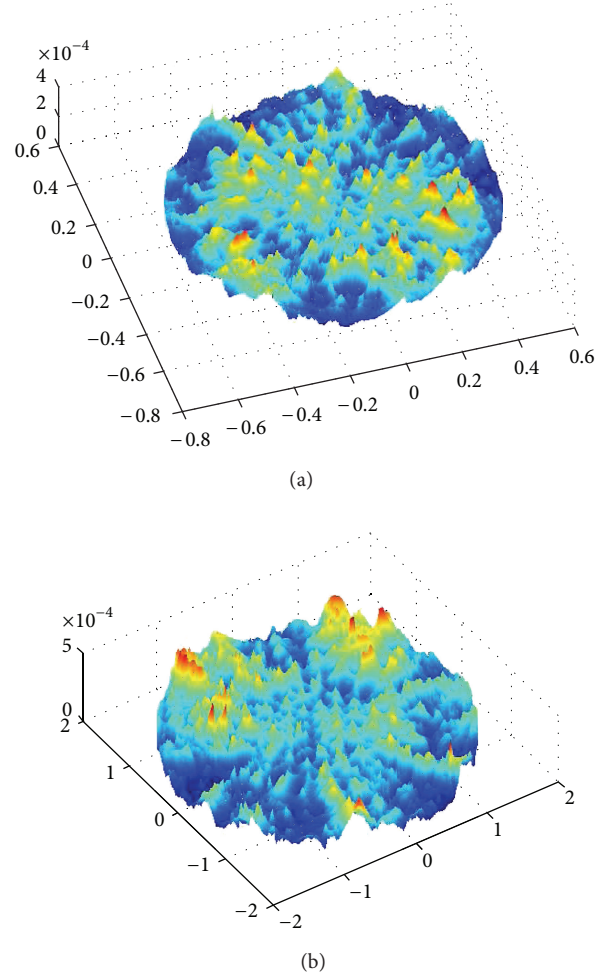


FIGURE 3: Error surface plots for reflector 1 (a) and reflector 2 (b) for the data of Table 3 with  $\kappa_{nn} = 2$ . For better visibility, the plots were obtained by interpolating the error at each mesh point to a continuous function. Note that the error distribution is relatively uniform, although slight spikes of the error tend to occur near the boundary.

problems arising in beam shaping problems, for example see [7–9].

As a proof of concept, we implemented both schemes using MATLAB and the commercial LP solver MOSEK and showed that the new scheme is easy to implement and makes

it possible to solve the problem on much finer meshes. In particular, we demonstrated that the number of constraints are greatly reduced by the new scheme, to as little as 0.09% of the full number of constraints. Our aim was to show the validity of the scheme, and we contend that performance could be further improved, for instance, by using a compiled computer language or a more specialized LP solver for transportation problems.

One drawback of the scheme is that it is quite slow. See Table 2. The most time-consuming part of the algorithm was not the solution of the LP problem but assembling the problem, that is, deciding which constraints to be included. We believe that this step, in particular, could be improved by using a compiled computer language instead of MATLAB.

There are a number of possible directions for future research. We did not rigorously prove that the scheme converges, although we strongly expect that it does. In fact, the decreasing error shown in Table 2 gives evidence for this. It would be valuable to have a rigorous proof of convergence. We also expect that PDE-based schemes for solving the transportation cost with quadratic costs [13, 17–19] could be adapted to the problem at hand, leading to faster algorithms.

## Appendices

### A. Details on Theorem 3

We give here some more details on Theorem 3 from Section 2. We use the same notation as in this section. This is a summary of definitions and results from [2].

In [2], the following notion of a reflector pair and its associated ray-tracing map is used.

*Definition A.1.* Let  $C_{>0}(\bar{D})$  denote the set of all continuous functions on  $\bar{D}$  whose range is contained in the interval  $(0, \infty)$ . A pair  $(\rho, z) \in C_{>0}(\bar{D}) \times C(\bar{T})$  is called a *reflector pair* if  $\hat{\rho}, \bar{z} > 0$  and

$$\begin{aligned} \hat{\rho}(\mathbf{m}) &= \sup_{\mathbf{x} \in \bar{T}} \left( \frac{1}{\bar{z}(\mathbf{x})} K(\mathbf{m}, \mathbf{x}) \right) \quad \forall \mathbf{m} \in \bar{D}, \\ \bar{z}(\mathbf{x}) &= \sup_{\mathbf{m} \in \bar{D}} \left( \frac{1}{\hat{\rho}(\mathbf{m})} K(\mathbf{m}, \mathbf{x}) \right) \quad \forall \mathbf{x} \in \bar{T}. \end{aligned} \tag{A.1}$$

Here, notations (4) and (3) were used.

*Definition A.2.* Let  $(\rho, z) \in C_{>0}(\bar{D}) \times C(\bar{T})$  be a reflector pair. Define its *reflector map*, or *ray-tracing map*, as a set-valued map  $\gamma : \bar{D} \rightarrow \{\text{subsets of } \bar{T}\}$  via

$$\gamma(\mathbf{m}) = \left\{ \mathbf{x} \in \bar{T} \mid \hat{\rho}(\mathbf{m}) = \frac{1}{\bar{z}(\mathbf{x})} K(\mathbf{m}, \mathbf{x}) \right\} \quad \text{for } \mathbf{m} \in \bar{D}. \tag{A.2}$$

In [2], it is shown that  $\gamma(\mathbf{m})$  is, in fact, single-valued for almost all  $\mathbf{m} \in \bar{D}$ . Therefore,  $\gamma$  may be regarded as a transformation  $\gamma : \bar{D} \rightarrow \bar{T}$ .

If  $(\rho, z)$  is a “reflector pair” in the above sense, the following can be shown [2], justifying the choice of nomenclature. If physical copies of the corresponding surfaces (1) are made from a reflective material, then a ray emitted from the origin in the direction  $\mathbf{m} \in \bar{D}$  will be reflected to a ray in the negative  $z$ -direction labeled by  $\mathbf{x} = \gamma(\mathbf{m}) \in \bar{T}$ .

With this, the reflector problem can be formulated rigorously.

*Problem A.3 (reflector problem).* For given input and output intensities  $I(\mathbf{m}), \mathbf{m} \in \bar{D}$ , and  $L(\mathbf{x}), \mathbf{x} \in \bar{T}$ , respectively, satisfying global energy conservation  $\int_{\bar{D}} I(\mathbf{m}) d\sigma = \int_{\bar{T}} L(\mathbf{x}) dx$ , find a pair  $(\rho, z) \in C_{>0}(\bar{D}) \times C(\bar{T})$  that satisfies the following conditions:

- (i)  $(\rho, z)$  is a reflector pair in the sense of Definition A.1;
- (ii) the ray-tracing map  $\gamma : \bar{D} \rightarrow \bar{T}$  satisfies

$$\int_{\gamma^{-1}(\tau)} I(\mathbf{m}) d\sigma = \int_{\tau} L(\mathbf{x}) dx \tag{A.3}$$

for any Borel set  $\tau \subseteq \bar{T}$ .

Here,  $d\sigma$  denotes the standard area element on the sphere  $S^2$ . The second condition is local energy conservation.

As we indicate below, the reflector problem can be reformulated in the following form.

*Problem A.4 (minimize the functional).*

$$\mathcal{F}(r, \zeta) = \int_{\bar{D}} r(\mathbf{m}) I(\mathbf{m}) d\sigma + \int_{\bar{T}} \zeta(\mathbf{x}) L(\mathbf{x}) dx \tag{A.4}$$

on the set  $\text{Adm}(\bar{D}, \bar{T}) = \{(r, \zeta) \in C(\bar{D}) \times C(\bar{T}) \mid r(\mathbf{m}) + \zeta(\mathbf{x}) \geq \log K(\mathbf{m}, \mathbf{x}) \text{ for all } \mathbf{m} \in \bar{D}, \mathbf{x} \in \bar{T}\}$ .

Problem A.4 is the dual problem to the problem of finding a measure-preserving map from  $\bar{D} \subseteq S^2$  to  $\bar{T} \subseteq \mathbb{R}^2$  which minimizes the transportation functional  $P \mapsto \int_{\bar{D}} K(\mathbf{m}, P(\mathbf{m})) I(\mathbf{x}) d\sigma$ . Problems A.3 and A.4 are equivalent, as expressed in the following theorem.

**Theorem A.5** (see [2]). *Let  $(\rho, z) \in C_{>0}(\bar{D}) \times C(\bar{T})$  be a reflector pair. Then  $(\log \hat{\rho}, \log \bar{z}) \in \text{Adm}(\bar{D}, \bar{T})$ . The following statements are equivalent:*

- (i)  $(\rho, z)$  solves the reflector Problem A.3.
- (ii)  $(\log \hat{\rho}, \log \bar{z})$  solves Problem A.4.

Thus, solving the reflector construction problem in Problem A.3 is equivalent to solving the infinite-dimensional LP problem in Problem A.4. Indeed, it can be shown that a solution as in Theorem A.5 exists. See Corollary 6.5 in [2].

## B. An Explicit Analytic Solution to the Reflector Problem

In order to obtain a data set where an explicit analytic solution is known, we consider a special given configuration of reflectors and then solve the “forward” problem of determining the output intensity this system produces for a given input intensity. This general family of solutions we obtained was used to generate the explicit synthetic data set in Section 4.1.

**B.1. Construction of a Pair of Reflectors  $R_1, R_2$ .** Consider the following set of two reflectors  $R_1$  and  $R_2$ , sketched in Figure 4. Let  $\mathbf{a} = (a, b, c)$  be a given point in  $\mathbb{R}^3$ , and let  $R > 0$  and  $\alpha > 0$  be two positive numbers with  $R > |\mathbf{a}| = \sqrt{a^2 + b^2 + c^2}$ . Now, let the first reflector  $R_1$  be the boundary of a prolate spheroid (ellipsoid of revolution) with foci at the origin  $\mathbf{O}$  and at  $\mathbf{a}$  and with major diameter  $R$ . Thus, for each point  $\mathbf{P}$  on  $R_1$ , the sum of the distances from each of the two foci  $\overline{\mathbf{PO}} + \overline{\mathbf{Pa}}$  equals  $R$ . Let the second reflector  $R_2$  be the boundary of a paraboloid whose main axis is the negative  $z$ -axis and whose focus is at  $\mathbf{a}$  with focal parameter  $2\alpha$ . Thus, for each point  $\mathbf{p} = (x, y, z)$  on  $R_2$ , the sum of the distance to the focus and the shifted  $z$ -component  $\overline{\mathbf{Pa}} + (z - c)$  equals  $2\alpha$ .

Note that by definition of the two reflectors, any cone of light rays emitted from the origin  $\mathbf{O}$  will be transformed into a beam of parallel rays traveling in the direction of the negative  $z$ -axis. Indeed, a ray emitted from  $\mathbf{O}$  will be reflected off  $R_1$  toward the focus  $\mathbf{a}$ . This ray will then be reflected in the direction of the negative  $z$ -axis by  $R_2$ . See again Figure 4.

It is not hard to find explicit expressions for the two reflectors. If we write  $R_1 = \{\rho(\mathbf{m}) \cdot \mathbf{m} \mid \mathbf{m} \in S^2\}$  and  $R_2 = \{(x, z(\mathbf{x})) \mid \mathbf{x} = (x, y) \in \mathbb{R}^2\}$ , then we have the following expressions:

$$\text{radial function } \rho \text{ for } R_1: \rho(\mathbf{m}) = \frac{R^2 - |\mathbf{a}|^2}{2(R - \mathbf{m} \cdot \mathbf{a})}, \quad (\text{B.1})$$

$$\text{equation for } R_2: -4\alpha q = |\mathbf{p}|^2 - 4\alpha^2, \quad (\text{B.2})$$

where we used the shifted coordinates  $(\mathbf{p}, q) = (x, y, z) - (a, b, c) = (\mathbf{x}, z) - \mathbf{a}$ .

**B.2. Ray-Tracing Map  $\gamma$ .** One can now determine the ray-tracing map  $\gamma$  corresponding to the reflector pair  $R_1, R_2$ . See again Figure 4.

Let  $\mathbf{m} \in S^2$  be a given direction. Thus,  $\mathbf{m} = (\mathbf{m}_x, m_z)$  with  $\mathbf{m}_x \in \mathbb{R}^2$  and  $|\mathbf{m}_x|^2 + m_z^2 = 1$ . To determine the image  $\gamma(\mathbf{m})$ , consider the ray emitted from the origin in the direction  $\mathbf{m}$ . The ray will encounter the first reflector at the point  $\rho(\mathbf{m}) \cdot \mathbf{m}$  and will then be reflected towards the second focus  $\mathbf{a}$ . Thus, the reflected ray can be parameterized by  $\mathbf{a} + \lambda \cdot (\mathbf{a} - \rho(\mathbf{m}) \cdot \mathbf{m})$ , where  $\lambda$  is a parameter. With (B.2), this yields that the point

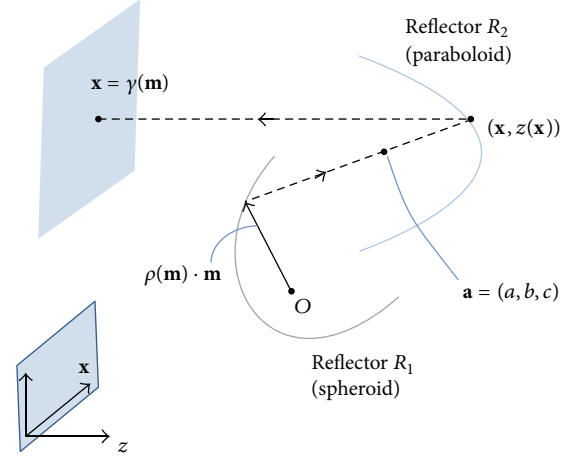


FIGURE 4: Sketch of the reflectors  $R_1$  and  $R_2$ .  $R_1$  is the boundary of an ellipsoid whose foci are the points  $\mathbf{O}$  and  $\mathbf{a}$ .  $R_2$  is the boundary of a paraboloid whose focus is  $\mathbf{a}$ . The sketch shows a two-dimensional cross section. A ray given by the direction  $\mathbf{m} \in S^2$  will be reflected by  $R_1$  and then by  $R_2$  to a ray in the direction of the negative  $z$ -axis.

where the reflected ray hits the second reflector is  $\mathbf{a} + \lambda^* \cdot (\mathbf{a} - \rho \cdot \mathbf{m})$ , where

$$\begin{aligned} \lambda^* &= \frac{2\alpha}{(R - \rho)^2 - (c - \rho m_z)^2} ((R - \rho) - (c - \rho m_z)) \\ &= \frac{2\alpha}{R + c - \rho(1 + m_z)}. \end{aligned} \quad (\text{B.3})$$

The projection of this point to the  $xy$ -plane is the value of the ray-tracing map. Thus, we have the following explicit formula for the ray-tracing map:

$$\begin{aligned} \gamma(\mathbf{m}) &= \begin{pmatrix} a \\ b \end{pmatrix} + \frac{2\alpha}{R + c - \rho(\mathbf{m})(1 + m_z)} \\ &\quad \times \left( \begin{pmatrix} a \\ b \end{pmatrix} - \rho(\mathbf{m}) \mathbf{m}_x \right), \quad \mathbf{m} = (\mathbf{m}_x, m_z) \in S^2. \end{aligned} \quad (\text{B.4})$$

**B.3. Solution of the Forward Problem.** We can now solve the forward problem. Given the reflector pair  $R_1, R_2$  as defined above, an input aperture  $\overline{\Omega} \subseteq S^2$  and an intensity distribution  $I(\mathbf{m})$ ,  $\mathbf{m} \in \overline{\Omega}$ , find the output aperture  $\overline{T} \subseteq \mathbb{R}^2$  and the output intensity  $L(\mathbf{x})$ ,  $\mathbf{x} \in \overline{T}$ .

The output aperture is simply the image of  $\overline{\Omega}$  under the ray-tracing map  $\gamma$ :  $\overline{T} = \gamma(\overline{\Omega})$ . To find the induced intensity  $L(\mathbf{x})$ , use the defining property

$$\int_{\omega} I(\mathbf{m}) d\sigma = \int_{\gamma(\omega)} L(\mathbf{x}) d\mathbf{x} \quad (\text{B.5})$$

for all Borel sets  $\omega \subseteq \overline{\Omega}$ . This is an energy balance equation.

The above integral equation allows us to find an explicit expression for  $L(\mathbf{x})$  given  $I(\mathbf{m})$  or vice versa. We consider, for simplicity, the case that  $\overline{\Omega}$  is contained in the left hemisphere

$S_-^2 = S^2 \cap \{(x, y, z) \in \mathbb{R}^3 \mid z < 0\}$ . We can then use coordinates:

$$\begin{aligned} \tau : \{(m_x, m_y) \in \mathbb{R}^2 \mid m_x^2 + m_y^2 < 1\} &\longrightarrow S_-^2, \\ (x, y) &\longmapsto (m_x, m_y, -\sqrt{1 - m_x^2 - m_y^2}). \end{aligned} \quad (\text{B.6})$$

In these coordinates, the standard measure on  $S_-^2$  is given by  $d\sigma = dm_x dm_y / |m_z|$ , where  $m_z = -\sqrt{1 - m_x^2 - m_y^2}$ . Using this, (B.5) is then equivalent to the equation

$$I(\mathbf{m}_x, m_z) \frac{1}{|m_z|} = L(\gamma(\tau(\mathbf{m}_x))) J(\gamma \circ \tau). \quad (\text{B.7})$$

Here,  $J(\gamma \circ \tau) = |\det(\partial(\gamma \circ \tau)/\partial m_x \quad \partial(\gamma \circ \tau)/\partial m_y)|$  denotes the Jacobian of the map  $\gamma \circ \tau$ . With the help of a computer algebra system like Mathematica, this can be evaluated explicitly as

$$\begin{aligned} J(\gamma \circ \tau) &= 4\alpha^2 (|\mathbf{a}|^2 - R^2)^2 \\ &\quad \times \left( -m_z \left( 2(c+R) \mathbf{m}_x \cdot \begin{pmatrix} a \\ b \end{pmatrix} \right. \right. \\ &\quad \left. \left. - (1+m_z) |(a, b)|^2 \right. \right. \\ &\quad \left. \left. - (c+R)^2 (1-m_z) \right)^2 \right)^{-1}, \end{aligned} \quad (\text{B.8})$$

where again  $m_z = -\sqrt{1 - m_x^2 - m_y^2}$ .

## Acknowledgment

Tilmann Glimm gratefully acknowledges the contribution of V. Oliker for the unpublished collaborative work on an algorithm similar to the one described in this paper for a different reflector design problem. This work was done during 2002 to 2004 at Emory University.

## References

- [1] V. Oliker, "Mathematical aspects of design of beam shaping surfaces in geometrical optics," in *Trends in Nonlinear Analysis*, M. Kirkilionis, S. Krömker, R. Rannacher, and F. Tomi, Eds., pp. 191–222, Springer, Berlin, Germany, 2002.
- [2] T. Glimm, "A rigorous analysis using optimal transport theory for a two-reflector design problem with a point source," *Inverse Problems*, vol. 26, no. 4, Article ID 045001, p. 16, 2010.
- [3] Y. Brenier, "Polar factorization and monotone rearrangement of vector-valued functions," *Communications on Pure and Applied Mathematics*, vol. 44, no. 4, pp. 375–417, 1991.
- [4] L. A. Caffarelli, "Allocation maps with general cost functions," in *Lecture Notes in Pure and Applied Mathematics*, pp. 29–35, Dekker, New York, NY, USA, 1996.
- [5] W. Gangbo and R. J. McCann, "Optimal maps in Monge's mass transport problem," *Comptes Rendus de l'Académie des Sciences*, vol. 321, no. 12, pp. 1653–1658, 1995.
- [6] C. Villani, *Optimal Transport: Old and New*, vol. 338 of *Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences]*, Springer, Berlin, Germany, 2009.
- [7] T. Glimm and V. Oliker, "Optical design of two-reflector systems, the Monge-Kantorovich mass transfer problem and Fermat's principle," *Indiana University Mathematics Journal*, vol. 53, no. 5, pp. 1255–1277, 2004.
- [8] T. Glimm and V. Oliker, "Optical design of single reflector systems and the Monge-Kantorovich mass transfer problem," *Journal of Mathematical Sciences*, vol. 117, no. 3, pp. 4096–4108, 2003.
- [9] X.-J. Wang, "On the design of a reflector antenna. II," *Calculus of Variations and Partial Differential Equations*, vol. 20, no. 3, pp. 329–341, 2004.
- [10] T. Graf, *On the near-field rector problem and optimal transport [Ph.D. thesis]*, Emory University, 2010.
- [11] V. Oliker, "Designing freeform lenses for intensity and phase control of coherent light with help from geometry and mass transport," *Archive for Rational Mechanics and Analysis*, vol. 201, no. 3, pp. 1013–1045, 2011.
- [12] E. Haber, T. Rehman, and A. Tannenbaum, "An efficient numerical method for the solution of the  $L_2$  optimal mass transfer problem," *SIAM Journal on Scientific Computing*, vol. 32, no. 1, pp. 197–211, 2010.
- [13] J.-D. Benamou and Y. Brenier, "A computational fluid mechanics solution to the Monge-Kantorovich mass transfer problem," *Numerische Mathematik*, vol. 84, no. 3, pp. 375–393, 2000.
- [14] G. L. Delzanno, L. Chacón, J. M. Finn, Y. Chung, and G. Lapenta, "An optimal robust equidistribution method for two-dimensional grid adaptation based on Monge-Kantorovich optimization," *Journal of Computational Physics*, vol. 227, no. 23, pp. 9841–9864, 2008.
- [15] G. Awanou, "Spline element method for the Monge-Ampere equation," December 2010.
- [16] L. Chacón, G. L. Delzanno, and J. M. Finn, "Robust, multidimensional mesh-motion based on Monge-Kantorovich equidistribution," *Journal of Computational Physics*, vol. 230, no. 1, pp. 87–103, 2011.
- [17] J.-D. Benamou, B. D. Froese, and A. M. Oberman, "Two numerical methods for the elliptic Monge-Ampère equation," *Mathematical Modelling and Numerical Analysis*, vol. 44, no. 4, pp. 737–758, 2010.
- [18] B. D. Froese and A. M. Oberman, "Fast finite difference solvers for singular solutions of the elliptic Monge-Ampère equation," *Journal of Computational Physics*, vol. 230, no. 3, pp. 818–834, 2011.
- [19] V. Zheligovsky, O. Podvigina, and U. Frisch, "The Monge-Ampère equation: various forms and numerical solution," *Journal of Computational Physics*, vol. 229, no. 13, pp. 5043–5061, 2010.
- [20] V. I. Oliker and L. D. Prussner, "On the numerical solution of the equation  $(\partial^2 z / \partial x^2)(\partial^2 z / \partial y^2) - ((\partial^2 z / \partial x \partial y))^2 = f$  and its discretizations. I," *Numerische Mathematik*, vol. 54, no. 3, pp. 271–293, 1988.
- [21] J. W. Barrett and L. Prigozhin, "A mixed formulation of the Monge-Kantorovich equations," *Mathematical Modelling and Numerical Analysis*, vol. 41, no. 6, pp. 1041–1060, 2007.
- [22] L. Rüschemdorf and L. Uckelmann, "Numerical and analytical results for the transportation problem of Monge-Kantorovich," *Metrika*, vol. 51, no. 3, pp. 245–258, 2000.

- [23] C. Canavesi, W. J. Cassarly, and J. P. Rolland, "Observations on the linear programming formulation of the single reflector design problem," *Optics Express*, vol. 20, no. 4, pp. 4050–4055, 2012.
- [24] C. Canavesi, W. J. Cassarly, and J. P. Rolland, "Implementation of the linear programming algorithm for freeform reflector design," p. 84850E-84850E-6, 2012.
- [25] X.-J. Wang, "On the design of a reflector antenna," *Inverse Problem*, vol. 12, no. 3, pp. 351–375, 1996.
- [26] R. Cools, "An encyclopaedia of cubature formulas," *Journal of Complexity*, vol. 19, no. 3, pp. 445–453, 2003.
- [27] P.-O. Persson and G. Strang, "A simple mesh generator in Matlab," *SIAM Review*, vol. 46, no. 2, pp. 329–345, 2004.
- [28] "The MOSEK optimization software," <http://www.mosek.com/>.
- [29] I. The MathWorks. Support pages for matlab, 2012, <http://www.mathworks.com/support/solutions/en/data/1-IHYHFZ/index.html>.





# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

