Freeform Reflector Design With Extended Sources

2010

Florian Fournier
University of Central Florida

Find similar works at: https://stars.library.ucf.edu/etd

University of Central Florida Libraries http://library.ucf.edu

Part of the Electromagnetics and Photonics Commons, and the Optics Commons

STARS Citation


This Doctoral Dissertation (Open Access) is brought to you for free and open access by STARS. It has been accepted for inclusion in Electronic Theses and Dissertations by an authorized administrator of STARS. For more information, please contact lee.dotson@ucf.edu.
FREEFORM REFLECTOR DESIGN
WITH EXTENDED SOURCES

by

FLORIAN FOURNIER
B.S. Institut d’Optique, 2004
M.S. University of Central Florida, 2008

A dissertation submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
in CREOL, the College of Optics and Photonics
at the University of Central Florida
Orlando, Florida

Summer Term
2010

Major Professor: Jannick P. Rolland
© 2010 Florian Fournier
Geometrical optics is either very simple, or else it is very complicated...  

— Richard P. Feynman in Lectures in Physics, Vol. I
ABSTRACT

Reflector design stemmed from the need to shape the light emitted by candles or lamps. Over 2,000 years ago people realized that a mirror shaped as a parabola can concentrate light, and thus significantly boosts its intensity, to the point where objects can be set afire. Nowadays many applications require an accurate control of light, such as automotive headlights, streetlights, projection displays, and medical illuminators. In all cases light emitted from a light source can be shaped into a desired target distribution with a reflective surface.

Design methods for systems with rotational and translational symmetry were devised in the 1930s. However, the freeform reflector shapes required to illuminate targets with no such symmetries proved to be much more challenging to design. Even when the source is assumed to be a point, the reflector shape is governed by a set of second-order partial non-linear differential equations that cannot be solved with standard numerical integration techniques. An iterative approach to solve the problem for a discrete target, known as the method of supporting ellipsoids, was recently proposed by Oliker. In this research we report several efficient implementations of the method of supporting ellipsoids, based on the point source approximation, and we propose new reflector design techniques that take into account the extent of the source.

More specifically, this work has led to three major achievements. First, a thorough analysis of the method of supporting ellipsoids was performed that resulted in two alternative implementations of the algorithm, which enable a fast generation of freeform reflector shapes within the point source approximation. We tailored the algorithm in order to provide control over
the parameters of interest to the designers, such as the reflector scale and geometry. Second, the shape generation algorithm was used to analyze how source flux can be mapped onto the target. We derived the condition under which a given source-target mapping can be achieved with a smooth continuous surface, referred as the integrability condition. We proposed a method to derive mappings that satisfy the integrability condition. We then use these mappings to quickly generate reflector shapes that create continuous target distributions as opposed to reflectors generated with the method of supporting ellipsoids that create discrete sets of points on the target. We also show how mappings that do not satisfy the integrability condition can be achieved by introducing step discontinuities in the reflector surface. Third, we investigated two methods to design reflectors with extended sources. The first method uses a compensation approach where the prescribed target distribution is adjusted iteratively. This method is effective for compact sources and systems with rotational or translational symmetry. The second method tiles the source images created by a reflector designed with the method of supporting ellipsoids and then blends the source images together using scattering in order to obtain a continuous target distribution. This latter method is effective for freeform reflectors and target distributions with no sharp variations.

Finally, several case studies illustrate how these methods can be successfully applied to design reflectors for general illumination applications such as street lighting or luminaires. We show that the proposed design methods can ease the design of freeform reflectors and provide efficient, cost-effective solutions that avoid unnecessary energy consumption and light pollution.
ACKNOWLEDGMENTS

The very topic of this dissertation, non-imaging optics, was new to me when I started my doctorate at the College of Optics and Photonics. It has been a long way since then! I would like to thank here all the people who have given me the opportunity to discover this field, which I have become passionate about.

First of all, I would like to thank my advisor, Prof. Jannick P. Rolland, for her guidance and support over the course of this PhD. She taught me confidence, perseverance, and helped me on numerous occasions to take a step back when I was getting a little too tangled in my research.

I feel much honored to have had Dr William J. Cassarly as my mentor. His expertise, patience and wisdom were invaluable. The many stimulating weekly phone conversations we have had over past three years have significantly shaped this work.

I would also like to thank Dr Shin-Tson Wu and Dr Winston Schoenfeld for serving on my committee. They are both inspiring professors and advisors.

This work would not have been possible without the support from Optical Research Associates, which provided a 3-year scholarship (2007-2010), as well as the educational license for the illumination design software LightTools. VDC Display Systems and the Office of Naval Research Virtual Technologies and Environments (VIRTE) funded my early work (2006-2007) and gave me the opportunity to start working in the field of illumination.
Working at the ODAlab has been a great adventure, and I was lucky to interact with engaging and talented people. In particular, I would like to acknowledge Ozan Cakmakci, Tobias Schmid, and Cristina Canavesi for all the stimulating discussions we had throughout this PhD.

My doctorate started in sunny Florida and ended in (mostly sunny – but quite snowy at times) Rochester. I would like to thank everyone who helped made the transition from the University of Central Florida to the University of Rochester easier. Many thanks to the staff at CREOL and at the Institute of Optics, in particular Rachel Franzetta and Gina Kern; their help and kindness were always greatly appreciated.

Finally, I want to acknowledge all the other people that have been supporting my work, each in their own way, including Kevin Thompson, Tina Kidger, Christopher Berrios, Sylvano do Nascimento, and above all my parents, who provided a constant supply of the French food I could not easily find in the United States!
# TABLE OF CONTENTS

LIST OF FIGURES ........................................................................................................... xii

LIST OF TABLES .................................................................................................................. xxii

LIST OF ACRONYMS .......................................................................................................... xxiii

CHAPTER ONE - INTRODUCTION ....................................................................................... 24

1.1 A brief history of reflector design ............................................................................. 24
1.2 Optimization and modern illumination design ......................................................... 29
1.3 Motivation .................................................................................................................. 31
1.4 Research summary .................................................................................................. 34
1.5 Dissertation outline ............................................................................................... 37

CHAPTER TWO - RELATED WORK .................................................................................... 39

2.1 Reflectors with rotational or translational symmetry ............................................... 39

2.1.1 Point source approximation .............................................................................. 39

2.1.2 Extended sources ............................................................................................. 48

2.2 Freeform reflectors ................................................................................................ 53

2.2.1 Equi-flux grid methods .................................................................................... 53

2.2.2 3D tailoring methods ....................................................................................... 55

2.2.3 Multiple surfaces ............................................................................................. 58

2.3 Faceted reflectors .................................................................................................... 59

2.4 Optimization techniques for reflector design ......................................................... 62
CHAPTER THREE - SHAPE GENERATION

3.1 Overview of the method of supporting ellipsoids ........................................ 67
3.2 Algorithm implementation ........................................................................ 74
   3.2.1 Algorithm A: Modified Kochengrin algorithm .................................... 74
   3.2.2 Algorithm B: Least squares optimization ............................................ 85
3.3 Target distribution evaluation .................................................................... 89
   3.3.1 Monte-Carlo estimation method ............................................................ 89
   3.3.2 Contour detection method ................................................................. 92
3.4 Surface representation and smoothing ....................................................... 94
   3.4.1 Surface representation ....................................................................... 94
   3.4.2 Surface interpolation ........................................................................ 98
3.5 Atlas of generated reflectors ....................................................................... 100
3.6 Alternative shape generation method using linear programming ............... 105

CHAPTER FOUR - SOURCE-TARGET MAPPING ............................................. 111

4.1 Mappings and shape generation ................................................................ 111
   4.1.1 Mappings for systems with rotational symmetry ............................... 111
   4.1.2 Mappings for systems without symmetry ......................................... 114
   4.1.3 Shape generation and slope mismatch .............................................. 118
   4.1.4 Wedge reflectors that minimize slope mismatch ............................... 120
4.2 The integrability condition ....................................................................... 123
4.3 Generating maps satisfying the integrability condition ............................. 130
   4.3.1 Map generation using the Oliker algorithm ....................................... 130
4.3.2 Map accuracy ........................................................................................................ 134
4.3.3 Some examples of integrable maps ......................................................................... 137
4.4 Generating smooth reflectors from maps ................................................................... 141
  4.4.1 Motivation ............................................................................................................. 141
  4.4.2 Effect of the integration path ................................................................................ 143
  4.4.3 Effect of the number of target points ................................................................. 146
  4.4.4 Examples of smooth reflectors generated from maps ......................................... 149
4.5 Evolution of maps with system parameters ............................................................... 155

CHAPTER FIVE - EXTENDED SOURCES ........................................................................... 161
5.1 Understanding the effects of extended sources ........................................................ 161
5.2 Compensation method for systems with rotational symmetry .................................. 165
  5.2.1 Overview of the iterative compensation algorithm .............................................. 165
  5.2.2 Target distribution evaluation ............................................................................. 168
  5.2.3 Adjusting the target distribution extent .............................................................. 170
  5.2.4 Calculating the correction factor ........................................................................ 172
  5.2.5 Case study: LED flashlight .................................................................................. 174
5.3 Tiling method for systems without symmetry ............................................................ 183
  5.3.1 Controlling the illuminance of the source images .............................................. 183
  5.3.2 Scaling source images ......................................................................................... 186
  5.3.3 Tiling source images ........................................................................................... 188
  5.3.4 Blending the source images with scattering ....................................................... 189
  5.3.5 Case studies ........................................................................................................ 190
CHAPTER SIX - SUMMARY AND CONCLUSIONS ................................................................. 194

APPENDIX A - COMPUTATION OF PHOTOMETRIC QUANTITIES ................................. 202

APPENDIX B - MONTE-CARLO SIMULATION ............................................................... 208

APPENDIX C - ELLIPSOID PARAMETERIZATION ......................................................... 216

APPENDIX D – DERIVATION OF MAPPING EQUATIONS BASED ON EQUI-FLUX GRID ................................................................................................................................. 219

LIST OF REFERENCES ..................................................................................................... 226
LIST OF FIGURES

Figure 1.1. Handheld lantern using a parabolic reflector, circa 1646. The flame of a candle is placed at the focus. The accompanying text explains that “even the smallest letters will be readable effortlessly at night” with this lantern. Reproduced from [2]. ........................................ 25

Figure 1.2. Early parabolic reflectors used in lighthouses. (a) Reflector made by William Hutchinson from small planar mirror facets [3]. (b) Section of a reflector from the Edinburgh Encyclopedia, and (c) arrangement of 7 similar reflectors to cover a 360° field of view [4]. ............................................................................................................ 25

Figure 1.3. Reflector designs from the late 19th century. (a) Luminaire using an elliptic reflector patented in 1874 [7]. (b) Streetlight reflector based on the revolution of a conic section [8]. ...................................................................................................................... 26

Figure 1.4. Design method described by Jolley in 1930. (Left) Flux curves used to map the source flux distribution to the desired target distribution. (Right) Graphical construction of the reflector shape based on the mapping established from the flux curves. Reproduced from [10]................................................................................................................... 27

Figure 1.5. Cities and light pollution. (Left) Light pollution halo over the city of Lancaster, California. (Right) Map of the artificial night sky brightness in Northern America. .......... 34

Figure 2.1. Geometry of a rotationally symmetric system. (a) Differential annulus-shaped solid angle used when calculating the cumulative source flux. (b) Differential ring-shaped area used when calculating the cumulative target flux for a near-field target...................... 41

Figure 2.2. Comparison of the cumulative flux distributions of the source and target in order to establish the relationship between the source ray angles θ and the target positions y. ........... 41

Figure 2.3. The reflector polar radius r, and the angles θ and β of the ray before and after reflection are linked through the traditional reflector equation (Equation 2.2). ................. 43

Figure 2.4. Reflector geometry with a near-field target................................................. 45

Figure 2.5. Reflector shapes for the four possible source-target mappings. While all mappings produce identical target distributions, they lead to very different reflector geometries. ...... 47
Figure 2.6. Effect of an extended source on the target distribution. The relative size of the source as seen from the reflector determines its impact on the target distribution. The closer the reflector is to the source, the larger the corresponding source image will be. 48

Figure 2.7. Equi-flux grids used to derive a mapping between the source and the target. 54

Figure 2.8. Uniform illumination with a faceted reflector. Facets along each wedge are tailored in order to illuminate the same region of the target. The superposition of similar patterns with different orientations produces uniform illuminance in the center and gradual edge falloff. 60

Figure 2.9. Faceted reflector used in an automotive headlamp (US Patent No 4,704,661). 61

Figure 2.10. Optimization techniques described by Vogl in 1972 (reproduced from [62]). (Left) In field patch mode, a grid of pinholes is placed in front of the reflector and the corresponding source images are used for analysis. (Right) In aperture flash mode, illuminance at a set of target points is evaluated by estimating the flashed area of the aperture at each point. 64

Figure 3.1. Discretization of the target illuminance distribution. 68

Figure 3.2. (a) Ellipsoids reflect all the rays emitted from one of their foci to the second focus. 69

Figure 3.3. Family of confocal ellipsoids. All ellipsoids have the same foci $F_1$ and $F_2$ but different focal parameters. The focal parameter acts as a scaling factor for the ellipsoids. 70

Figure 3.4. Definition of ellipsoids intersection rules. A rule is defined to determine how to generate the reflector surface from the combination of all ellipsoids. In this example, the first intersected ellipsoid reflects the ray to its corresponding target point. The ray on the left will therefore be reflected by $E_i$ at $I_i$ and hit target point $T_i$. 70

Figure 3.5. Crossing reflector geometries are obtained when using the set of ellipsoid facets the closest to the source. 72

Figure 3.6. Non-crossing reflector geometries are obtained when using the set of ellipsoid facets the farthest away from the source. 72

Figure 3.7. 3D view of crossing and non-crossing reflectors made of 25 ellipsoid facets. 73

Figure 3.8. Ellipsoid scaling process. By decreasing the focal parameter $d_i$, ellipsoid $E_i$ gradually transitions from collecting 0% of the flux to collecting 100% of the flux. One value of the parameter $d_i$ thus yields the desired amount of flux at target point $T_i$. 74
Figure 3.9. Flowchart of the algorithm based on the method of supporting ellipsoids. .............. 75

Figure 3.10. Choice of a reference ellipsoid to set the reflector size. With a crossing reflector each ellipsoid facet is located on the opposite side from its corresponding target point. ..... 77

Figure 3.11. Comparison of the peak error in the reference ellipsoid vs. all other ellipsoids during an algorithm run (log scale). Error is defined in Eq. 3.7 as the absolute value of the difference between measured target value and desired target value. .............................................. 81

Figure 3.12. Comparison of error distribution among ellipsoids before and after an algorithm run. The reference ellipsoid originally has the largest error. After convergence, error is randomly distributed and is below the Monte-Carlo statistical noise limit. .............................................. 81

Figure 3.13. Evolution of the values of the focal parameters during an algorithm run. .......... 82

Figure 3.14. Evolution of the normalized RMS error during an algorithm run for various increment scaling factors $\kappa$, and for no increment scaling (original unmodified algorithm). Increment scaling dramatically improves convergence speed. .............................................. 83

Figure 3.15. Relative convergence time vs. number of target points. .............................................. 85

Figure 3.16. Comparison of the convergence speed of the two proposed shape generation algorithms. In this example 9 target points were used. .............................................. 87

Figure 3.17. Effect of multi-threading on computation time. .............................................. 91

Figure 3.18. Contour detection method. From an interior point, we move until we find the facet edge and then proceed along the facet edge using an iterative bisection search at each step. .............................................. 93

Figure 3.19. Generation of the reflector surface from a set of 9 ellipsoids .............................................. 95

Figure 3.20. Exported reflectors made of (a) a triangular mesh and (b) a NURBS surface. ....... 96

Figure 3.21. Target distributions obtained with various surface sampling accuracies for a uniform 10×10 target grid. The sampling grid accuracy in spherical coordinates is indicated as number of points along $\theta \times$ number of points along $\phi$. .............................................. 97

Figure 3.22. Target distributions produced by interpolation of the polar radius of the reflector surface, for a uniform square target. .............................................. 99
Figure 3.23. Effect of the reflector geometry (crossing / non-crossing) on the reflector shape. 101

Figure 3.24. Effect of the target size on the reflector shape. .............................................................. 101

Figure 3.25. Effect of the target aspect ratio on the reflector shape. .................................................. 102

Figure 3.26. Example of a tilted target with non-constant z coordinates............................................. 102

Figure 3.27. Effect of a shift of the target along the x-axis (off-axis target). ................................. 103

Figure 3.28. Effect of the collection angle on the reflector shape (crossing geometry). ............. 103

Figure 3.29. Crossing and non-crossing reflector shapes for a wall-washer application. ........ 104

Figure 3.30. Example of a reflector shape that generates the letters “UCF”. ............................. 104

Figure 3.31. Self-blockage of the reflected rays by the reflector. The rays in red are reflected a second time by the reflector and thus do not reach their expected target location. This situation often arises with large targets and large collection angles......................................................... 105

Figure 3.32. Example of a set of reflector points generated with the variational approach. In this example a 3×3 target point grid was used in conjunction with a 15×60 ray grid. The source is isotropic and the reflector has a crossing geometry................................................................. 110

Figure 4.1. Source-target mapping for a system with rotational symmetry............................ 112

Figure 4.2. Geometrical construction of a reflector curve in order to achieve a specified source-target mapping. ........................................................................................................................................... 113

Figure 4.3. Spherical coordinate system angle conventions................................................................. 114

Figure 4.4. Elmer’s method [15] (extended to 3D) for the generation of reflector curves........ 115

Figure 4.5. Source-target mapping derived using symmetry considerations............................. 116

Figure 4.6. Generation of the reflector shape based on a source-target mapping for a uniform square target. Profile rib curves are generated at various angles $\phi$ (green contours). The rib curves are then interpolated into a smooth reflector using a NURBS surface. The target illuminance produced by the generated reflector departs significantly from the expected uniform square the reflector was originally designed for................................................................. 119
Figure 4.7. Slope mismatch arising during reflector generation. When the reflector surface is interpolated from a set of profiles along \( \theta \), the component of the surface normals along \( \phi \) may not match initial assumptions.

Figure 4.8. Wedge generation method that minimizes slope errors by integrating successively along \( \theta \) and \( \phi \). The green curve corresponds to integration from the reflector apex along \( \theta \) and the blue curves correspond to integration along \( \phi \). The set of generated points is then interpolated with a NURBS surface into a smooth wedge, as shown on the right.

Figure 4.9. Wedged reflectors and their corresponding target illuminance distributions. Wedged reflectors avoid slope mismatch arising when trying to generate a smooth reflector shape from the mapping shown in Figure 4.5. Generating more wedges minimizes artifacts in the target distribution but can make it difficult to manufacture the shape accurately.

Figure 4.10. (a) Source-target mapping derived with the Oliker algorithm and satisfying the integrability condition. (b) Source-target mapping that does not fulfill the integrability condition derived from symmetry considerations.

Figure 4.11. Magnitude of \( \eta \) for the two mappings presented in Figure 4.10. \( \eta \) is plotted as a function of the target map location. The target map on the right results in large residual curl as we move away from the axis. In comparison, the integrable mapping on the left minimizes \( \eta \) over the entire target and can therefore be achieved with a smooth continuous reflector surface.

Figure 4.12. Reflectors generated from the source-target mapping depicted in Fig. 4.5 (using the Oliker algorithm) and their corresponding illuminance distributions.

Figure 4.13. The integrable map for a uniform square target maps a ray fans at various \( \phi \) angles onto curves on the target, except along the axis of symmetry of the system.

Figure 4.14. Map generation process. (a) Ray directions aimed towards the center of each facet are mapped to their corresponding target point. In order to preserve edges, points at the edge of the target are mapped to the ray directions corresponding to the middle of the outer facets instead of their center. (b) The discrete mapping relationships \( x = f(\theta, \phi) \) and \( y = g(\theta, \phi) \) can then be approximated by a surface to get a continuous mapping function. (c) The generated map plots the contours corresponding to constant \( \theta \) and \( \phi \) values.

Figure 4.15. Scaling of target values for maps. (a) Edge and corner target values must respectively be scaled by 0.5 and 0.25 to account for the fact that flux only needs to be
spread inward. (b) Target distribution when no scaling is applied (uniform target values). (c) Target distribution when edges and corners are scaled. ................................................................. 132

Figure 4.16. Approximation of discrete mapping data with Gridfit. Spherical coordinates are wrapped onto a rectangular grid and the $\phi$ range is extended to emulate periodicity of the data. .................................................................................................................. 134

Figure 4.17. Mapping integrability and flux mapping error for a uniform square target. ........ 136

Figure 4.18. Target maps for three different system geometries: square, rectangle and off-axis square. In all cases the target is uniform, the point source is isotropic, and the collection angle of the reflector is a hemisphere. ................................................................. 137

Figure 4.19. Map contours fitted with Lamé curves. The two Lamé curve parameters $a$ and $n$ are plotted on the right as functions of the source angle $\theta$ ($1/n$ is plotted since $n$ goes to infinity as the contour shape becomes a square). .................................................................................. 139

Figure 4.20. Integrable maps for reflector facets with different azimuth angles $\phi$ (0°, 22.5° and 45°) and a uniform square target. In all cases the elevation angle $\theta$ is 60° and the collection angle subtended by each facet is 20° by 20°. 225 ellipsoids were used to generate the maps. .................................................................................................................. 140

Figure 4.21. Direct fitting of the ellipsoids generated with the Oliker algorithm. We used the same set of data points used to generate the maps. Edge artifacts proved to be difficult to minimize, even when the number of ellipsoids increases. ......................................................................... 142

Figure 4.22. Generation of a smooth reflector using the Oliker algorithm.......................... 143

Figure 4.23. Effect of the integration path on the target illuminance distribution. Longer integrations paths produce more artifacts in the target distribution because residual map errors accumulate during integration, which translates into slope mismatch in the reflector surface.................................................................................................................. 145

Figure 4.24. Effect of the integration path on the target distribution for a reflector facet. Similarly to the example shown in Figure 4.23, longer integrations paths produce more artifacts in the target distribution because residual map errors accumulate during integration, which translates into slope mismatch in the reflector surface. ................................................................. 146

Figure 4.25. Effect of the number of ellipsoids in the initial reflector on the target distribution, for a uniform square target. The statistical noise for these simulations is 0.5%. The reflector
that produces the uniform target distribution on the right was generated in about 12 s on a 2.5 GHz Intel Quad core computer from an initial reflector made of 441 ellipsoids. ....... 147

Figure 4.26. Sag difference between the initial reflector surface made of 149 ellipsoids and the smooth reflector surface generated from its corresponding integrable map. The peaks in the plot correspond to the dips in the original reflector surface where multiple ellipsoid facets meet and slope discontinuities occur. ................................................................. 149

Figure 4.27. Performance of a reflector generated from a uniform rectangular map. ............ 150

Figure 4.28. Performance of a reflector generated from an off-axis uniform square target map. ................................................................................................................................. 151

Figure 4.29. Performance of a reflector generated from a Gaussian square target map. ........ 152

Figure 4.30. Target distribution error for the Gaussian square target in Figure 4.29. .......... 152

Figure 4.31. Performance of reflector facets generated from uniform square maps. .......... 153

Figure 4.32. Comparison of integrable maps for crossing and non-crossing reflector geometries. ......................................................................................................................... 155

Figure 4.33. Comparison of source-target mappings for an isotropic source and a Lambertian source. In both cases the target is a uniform square, and light is collected over a 120º cone. ............................................................. 156

Figure 4.34. Effect of the target size on the target map. The target is located at $z = 30$ ft, and its size varies from $30 \times 30$ ft to $240 \times 240$ ft. The map variations are plotted as a displacement of the map nodes relative to the $30 \times 30$ ft reference target map on the left. The blue arrows were magnified 4 times for clarity. The corresponding reflector shape is shown underneath each map (the distance between the point source and the reflector apex is the same in each case). ......................................................................................................................... 157

Figure 4.35. Effect of the target aspect ratio on the target map. The target is located at $z = 30$ ft, and its size varies from $60 \times 60$ ft to $240 \times 60$ ft. The point source is isotropic and light is collected over a hemisphere. All reflectors are about 1 ft in diameter. ................................................. 159

Figure 4.36. Effect of the reflector size on the target map. The map variations are plotted as a displacement of the map nodes relative to the reference map of the 1 ft diameter reflector. The target is a uniform $60 \times 60$ ft square and is located at $z = 30$ ft. The point source is isotropic and light is collected over a hemisphere. The output diameter of the reflector
varies from 0.1 ft to 100 ft. Each map thus corresponds to an order of magnitude difference in reflector scale compared with its adjacent maps. ................................................................. 160

Figure 5.1. Intensity distribution of an LED and apparent source area in several directions. For a Lambertian source, the larger the projected area of the source, the higher the intensity.... 162

Figure 5.2. Pinhole camera images of the source produced by various points on the reflector. 163

Figure 5.3. Limitations of single reflectors with extended sources. In this example, all rays coming from the centroid of the source are focused to a point at the center of the target. The corresponding reflector shape is an ellipsoid. Once the mapping has been set, no degrees of freedom are left to control the size and orientation of the source images. ...................... 164

Figure 5.4. Flowchart of the iterative compensation algorithm........................................... 167

Figure 5.5. Illustration of the tradeoff between the measured target point density and the statistical noise inherent to Monte Carlo ray tracing. ......................................................... 169

Figure 5.6. The extent of the prescribed distribution can be adjusted so that the falloff of the measured output distribution matches the falloff of the target distribution...................... 171

Figure 5.7. Correspondance between the density of the rays on the target and the prescribed target distribution. Regions with high prescribed illuminance correspond to high ray densities................................................................................................................................. 171

Figure 5.8. System geometry of the LED flashlight case study. The system is made of a flat emitter and a rotationally symmetric reflector. As a constraint, the aperture diameter must be equal to 9.5 mm. In this case $\alpha_{\text{min}} = 90^\circ$, $\alpha_{\text{max}} = 150^\circ$, $\beta_{\text{max}} = 22^\circ$, and $\beta_{\text{min}} = -22^\circ$ ........... 175

Figure 5.9. Measured output intensity distribution as the diameter of the source increases. The table on the right indicates the maximum apparent source size as seen from the reflector in each case, as well as the half-angle of the output distribution in the étendue-limited case (the étendue limit is calculated for the light collected by the reflector).................................. 176

Figure 5.10. Apparent source size in degrees as seen from the reflector for various output angles. ................................................................................................................................. 177

Figure 5.11. Evolution of the MF values during the iterative process. In all cases it takes about 20 cycles to reach a minimum. ............................................................. 177

xix
Figure 5.12. (Left) Measured intensity distributions and (Right) corresponding prescribed intensity distributions after 25 iterations of the compensation algorithm.

Figure 5.13. (Left) Comparison of the shapes of the reflector before and after the iterative process for a 1 mm diameter source. (Right) For each Z coordinate the difference in Y coordinates of the initial and final reflector is plotted.

Figure 5.14. (Left) Comparison of the intensity patterns of the original Lambertian emitter and of the side emitter use in this simulation. (Right) Comparison of the output distributions with the new side emitter before and after the iterative process. The source diameter is 1 mm this case.

Figure 5.15. (Left) Measured output intensity distributions and (Right) prescribed distribution before and after the iterative process. In this case we use a 1 mm diameter source and the target is a uniform intensity distribution up to 10°.

Figure 5.16. (a) Single convergent reflector obtained with a 60×60 ft uniform target grid placed 30 ft away from the reflector. 10×10 target points were used. Target illuminance (b) with a point source, (c) with a cylindrical source and the standard algorithm, (d) with a cylindrical source and the PSA-based algorithm. All illuminance maps are normalized by the peak illuminance. 200×200 bins were used and 2.5 million rays were traced.

Figure 5.17. Illuminance at a given target point can be computed from the projected solid angle of the beam illuminating that point. When sources images do not overlap, each target point is simply illuminated by its corresponding ellipsoid patch.

Figure 5.18. Evolution of the target illuminance as the size of the source increases. Ideally, the target grid and the reflector shape should be adjusted so that source images tile the target area. 120×120 bins were used and 2.5 million rays were traced.

Figure 5.19. Optimization of the target grid in order to minimize gaps and overlap.

Figure 5.20. Effect of scattering on an illuminance distribution. Gaussian scattering with various spread angles $\sigma$ is applied uniformly across the reflector surface. 120×120 bins were used and 2.5 million rays were traced.

Figure 5.21. Reflector specifications and final crossing reflector design. The source is a Lambertian cylindrical surface source, 18 mm long and with a 2:1 aspect ratio. The final reflector has a maximum diameter of 33.5 cm.
Figure 5.22. Illuminance distribution on the target. The dotted line represents the limits of the 120×60 ft target area for which the design was optimized. 5 million rays were traced and 80×40 bins were used.

Figure 5.23. Reflector specifications and final non-crossing reflector design. The source is a 1×1 mm Lambertian square surface source. The final reflector has a maximum diameter equal to 23 mm and is 16 mm deep.

Figure 5.24. Illuminance distribution on the target. The dotted line represents the limits of the 50×50 cm target area for which the design was optimized. 5 million rays were traced and 60×60 bins were used.
LIST OF TABLES

Table 2.1. Flux conservation equations for various reflector geometries.............................. 43
Table 2.2. Effect of a cylindrical source on the target distribution as its size increases............ 49
Table 2.3. Summary of design techniques for systems with rotational / translational symmetry 52
Table 2.4. System of equations governing the general reflector problem .................................. 56
Table 3.1. Comparison of algorithm A and algorithm B. ......................................................... 88
Table 5.1. Standard deviation of the departure of the final measured intensity distribution from the target. As the source diameter increases, it becomes more difficult to closely match the target. ........................................................................................................................................ 179
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAD</td>
<td>Computer-Aided Design</td>
</tr>
<tr>
<td>CPC</td>
<td>Compound Parabolic Concentrator</td>
</tr>
<tr>
<td>CSG</td>
<td>Constructive Solid Geometry</td>
</tr>
<tr>
<td>HID</td>
<td>High Intensity Discharge</td>
</tr>
<tr>
<td>LED</td>
<td>Light-Emitting Diode</td>
</tr>
<tr>
<td>MF</td>
<td>Merit Function</td>
</tr>
<tr>
<td>NURBS</td>
<td>Non-Uniform Rational B-Spline</td>
</tr>
<tr>
<td>PSA</td>
<td>Projected Solid Angle</td>
</tr>
<tr>
<td>RMS</td>
<td>Root Mean Square</td>
</tr>
<tr>
<td>SA</td>
<td>Solid Angle</td>
</tr>
<tr>
<td>SMS</td>
<td>Simultaneous Multiple Surfaces</td>
</tr>
<tr>
<td>TED</td>
<td>Tailored Edge-ray Design</td>
</tr>
<tr>
<td>TIR</td>
<td>Total Internal Reflection</td>
</tr>
</tbody>
</table>
CHAPTER ONE - INTRODUCTION

This chapter introduces the art and science of reflector design and outlines the work presented in this dissertation. We start in Section 1.1 with a historical perspective on the field of non-imaging optics, more specifically with the development of reflector design since the 18th century. Then, we discuss in Section 1.2 the role of optimization in the design process with regard to today’s computational capabilities. The final sections of this chapter motivate this research and outline the organization of this dissertation.

1.1 A brief history of reflector design

Non-imaging optics is not a new science. The capacity of mirrors to concentrate light so that objects can be set afire has been known for centuries. The Arabian physicist Ibn Sahl gives a detailed written account of parabolic mirrors used for light concentration in the 10th century, in his treatise on burning mirrors and lenses [1]. It has been suggested that the high concentrating power of parabolas may have been known since antiquity. Indeed, the Greek mathematician Diocles described the focal properties of the parabola in the 2nd century BC; and legends claim that Archimedes set invading Roman ships afire during the siege of Syracuse in 212 BC by focusing sunrays with large parabolic mirrors. The use of parabolic mirrors for beam shaping, instead of concentration, only became common much later, at the end of the 18th century. In 1646, the treatise *Ars Magna Lucis et Umbrae* details the construction of a lantern using a parabolic mirror and a candle. This early flashlight prototype is reproduced in Figure 1.1.
Figure 1.1. Handheld lantern using a parabolic reflector, circa 1646. The flame of a candle is placed at the focus. The accompanying text explains that “even the smallest letters will be readable effortlessly at night” with this lantern. Reproduced from [2].

Figure 1.2. Early parabolic reflectors used in lighthouses. (a) Reflector made by William Hutchinson from small planar mirror facets [3]. (b) Section of a reflector from the Edinburgh Encyclopedia, and (c) arrangement of 7 similar reflectors to cover a 360° field of view [4].

It is unclear whether such lanterns were manufactured and used at the time. However, we know that parabolic mirrors were commonly used in lighthouses in the 19th century. In a lighthouse, the use of a parabolic reflector greatly intensifies the light beam in one direction so that boats can see it from further away. The Hunstanton lighthouse in England claims to be the first being equipped with a parabolic reflector in 1776. William Hutchinson, a dock master of Liverpool,
produced some of these early parabolic mirrors for lighthouses and described their construction. At the time, they were made of small facets of silvered glass set in a plaster cast molded to a paraboloid form, as shown in Figure 1.2a [3, 5]. It took some improvement in manufacturing techniques to be able to make smooth parabolic reflectors. Figure 1.2b shows an example of a smooth metallic reflector used in a lighthouse circa 1830 [4].

With the advent of the Argand lamp in 1780, which provided steady illumination, most lighthouses in Europe started being equipped with parabolic reflectors. A variation of the Argand lamp was made popular in the US around 1810 by the ship captain Winslow Lewis, although it seems that the exact reflector shape he used was rather spherical, leading to poor system efficiency. Nonetheless, it would soon equip all lighthouses in the US. However, because of the difficulty to produce large parabolic mirrors and their costly maintenance (heat and corrosion were the main culprits), the prismatic lens invented in 1823 by Fresnel and Arago would rapidly replace mirrors in lighthouses [6].

Figure 1.3. Reflector designs from the late 19th century. (a) Luminaire using an elliptic reflector patented in 1874 [7]. (b) Streetlight reflector based on the revolution of a conic section [8].
The use of conic reflectors for other lighting tasks became widespread towards the end of the 19th century. US patents can be found where conic shapes are used in luminaires and street lights, as shown in Figure 1.3a [7, 8]. Incremental steps in sophistication include using different pieces of conics in order to split the beam in different directions, or sweeping a conic shape along a curve to widen the beam along one direction, as shown in Figure 1.3b. It seems that this need originated from the desire to avoid glare from headlights, as described in a US patent in 1924 [9]. Faceted reflectors appear around the same time as a mean to improve beam uniformity.

![Figure 1.3](image)

**Figure 1.3.** The use of conic reflectors for other lighting tasks became widespread. US patents can be found where conic shapes are used in luminaires and street lights, as shown in Figure 1.3a [7, 8]. Incremental steps in sophistication include using different pieces of conics in order to split the beam in different directions, or sweeping a conic shape along a curve to widen the beam along one direction, as shown in Figure 1.3b. It seems that this need originated from the desire to avoid glare from headlights, as described in a US patent in 1924 [9]. Faceted reflectors appear around the same time as a mean to improve beam uniformity.

The major drawback of conic shapes is that they do not provide control of the flux over a continuum of directions. The transition from conic shapes to numerically integrated shapes allowing better flux control happened in the early 1930s. Jolley describes in 1930 a geometrical design method based on constructing successive reflector points [10, 11]. Figure 1.4 reproduces two of the design steps described by Jolley. However, no equation governing the reflector shape...
is given then. Boldyrev derives the general reflector equation in 1932 [12]. A patent filled in 1934 by Frank Benford on behalf of General Electric contains the traditional reflector equation for rotationally symmetric reflectors, and several examples of its application, as well as a technique for the numerical integration of the equation [13].

From the late 1930s to the early 1970s, there is a steady increase in patent publications related to illumination systems using reflectors, but it seems that no fundamentally new design techniques were devised, although more complex systems using hybrid shapes appear. In parallel, the field of solar concentration sparked an interest for a category of devices that achieve high collection efficiency. The most famous of them, the compound parabolic concentrator (CPC), appeared in the mid-1960s [14]. In 1974, Elmer publishes *The Optical Design of Reflectors*. It synthesizes the current knowledge and practice in the field and establishes a more rigorous approach to reflector design [15]. At the time, design techniques would use the point source approximation and would be limited to geometries with rotational or translational symmetry. Large sources and asymmetric distributions would be handled by empirical means and trial and error.

In the early 1990s, the concept of edge rays used for the design of solar concentrators is applied to the design of luminaires with extended Lambertian sources. Most of the methods focus on trough reflectors using strip or tubular sources [16, 17]. Another area of investigation is the use of multiple surfaces to circumvent the limitations inherent to single reflectors, which has lead to the Simultaneous Multiple Surface method (SMS) [18]. The method was originally devised for concentrators but since then has been investigated for luminaire design and even imaging systems. Since 2000, two major breakthroughs have occurred. First, two numerical solutions to
the general reflector equation have been reported in the literature: by Oliker [19], and by Ries and Muschaweck [20]. These methods will be discussed in more details in Chapter 2. Second, the ever-growing computational power has enabled the optimization of freeform surfaces for illumination systems.

1.2 Optimization and modern illumination design

While optimization algorithms have occasionally been used to improve the performance of non-imaging systems since the 1970s, optimization only started being part of illumination design software since 2000 [21, 22]. So why did it take so long for optimization to make its way into the world of illumination, given that lens designers had already used it for more than three decades? Let’s examine the three major reasons.

First, analyzing illumination systems requires much more computational power than imaging systems. While aberrations can be evaluated from a few dozens of carefully sampled rays across the pupil, illumination systems often require several hundreds of thousands of rays, if not several millions, in order to be accurately sampled. For instance, let’s consider a surface divided into 20×20 bins where we want to evaluate the illuminance. In order to obtain less than 5% statistical error with Monte-Carlo ray tracing, at least 160,000 rays must be traced (see Appendix C for more details on Monte Carlo ray tracing). If the system has losses such as absorption or scattering, even more rays will need to be traced in order to obtain the required accuracy.

Second, non-imaging systems are usually non-sequential. In imaging systems we know the sequence of surfaces a ray is going to hit, so optical software codes take advantage of this
knowledge to speed up computations. Non-imaging software allows rays to interact with any surface in any order, in a non-sequential fashion. This is computationally intensive, since we need to test the intersection of each ray with many surfaces, which often have complex geometries (spherical surfaces are not the norm).

Lastly, and most importantly, non-imaging systems are challenging to set up properly as optimization problems. As opposed to lens design, the merit function assessing the system performance is not unique. The goal of an optimization task could be maximizing collected flux, improving spatial uniformity, or obtaining a given color, among many others [22]. While both imaging and non-imaging systems have implicit constraints (lenses should not intersect with each other, for instance), these constraints are more difficult to handle in a non-imaging system because each entity in the system can serve a variety of purposes, and the rays can follow multiple paths. Parameterization of surfaces may be the most challenging problem facing non-imaging optics today. Most non-imaging systems use aspheric or segmented surfaces that cannot be described with a few parameters. Freeform surfaces represented by NURBS require dozens of control points and knot vectors to ensure good control over the surface shape [23]. This means that the parameter space to explore during optimization can be very large (as a side remark, freeform surfaces such as NURBS were originally devised for mechanical design so it should come as no surprise that they do not always suit optical problems very well). To make things worse, merit functions based on Monte-Carlo ray tracing are noisy, and tend to exhibit many local minima.
The combination of these three factors makes optimization with traditional approaches of many illumination systems beyond the reach of today’s computers. For that matter, two conditions appear to be necessary for the successful optimization of non-imaging systems. First, surfaces must be re-parameterized in order to reduce the number of effective optimization variables and the corresponding parameter space. Second, we need a good starting point that is reasonably close to the final solution. We explore these two aspects in the next chapters.

1.3 Motivation
Current reflector design techniques rely on simplified source models and tedious trial and error processes. This dissertation proposes a methodology for the design of reflectors with extended sources (as opposed to point sources). Specifically, our research addresses the design of reflector shapes that convert light emitted by an extended source into any desired target distribution, with or without symmetry. It is assumed that we know all relevant characteristics of the source, in particular its luminance distribution. Our goal is to obtain either a given intensity distribution (far-field target) or illuminance distribution (near-field target). No wave effects are taken into account; the geometrical optics approximation is always assumed. We will only use incoherent sources and the scale of the components we deal with makes diffraction effects negligible. Additionally, we limit our study to single reflective surfaces. By single surface, we include surfaces with possible slope or step discontinuities. The only requirement is that each ray emitted by the source must interact only once with the reflector surface, or hit the target directly. This constraint should not be perceived as overly restrictive. A wide range of applications in medical, automotive and general lighting have successfully used single reflectors for decades, and much
remains to be explored. In fact, part of this work focuses on better understanding the limitations of single surfaces, so that multiple surfaces can be used when they are relevant.

The apparent simplicity of a single surface is misleading. Nowadays, manufacturing techniques are capable of creating sophisticated freeform surfaces with precision. In fact, because tolerances are typically looser for illumination systems, reflectors have long been designed with aspheric shapes. In non-imaging optics, the factor limiting performance is often not manufacturing, but the design techniques. As we shall see in Chapter 2, a variety of design methods have been developed for 2D systems, i.e. for systems whose symmetry allows the problem to be analyzed in a 2D plane (such as rotational or translational symmetry). The transition from 2D to 3D systems with no symmetry is substantial. In fact, while design techniques for 2D systems were devised in the 1930s, the first practical 3D design methods only appeared in 2002! Because the problem is mathematically challenging, several assumptions need to be made about the source, such as assuming that the source is punctual or Lambertian. In most applications, the point source approximation is not acceptable: the extent of the source can have a tremendous impact on the target distribution, especially for systems with stringent compactness requirements. Also, the opto-mechanical structure of lamps often creates artifacts in the light emission pattern, so assuming an isotropic or Lambertian source significantly departs from real sources. Even within these approximations, obtaining a reflector shape can still be difficult and time-consuming.

When considering extended sources, there is no guarantee that a solution fulfilling the design requirements exists. There are often more constraints than degrees of freedom in the system, so tradeoffs must be made. As in lens design, a common approach is to find a good starting point,
and then use optimization. However, in contrast with lens design, there is no equivalent of aberration theory in non-imaging optics to guide the design process. It is therefore challenging to correlate results with specific features of the shape. Optimization essentially becomes a blind process, mostly based on lengthy trial and error and design experience. But for specific systems, such as single reflectors, it becomes possible to develop some insight. It is precisely the goal of this work to find an efficient parameterization of the reflector design problem that can provide more insight to the designer and make optimization more efficient. As of today, no technique is adapted for the design of reflectors with “real” non-Lambertian extended sources, and sheer computational power is not effective. In most cases, extensive manual tweaking and trial and error is still the norm.

Good reflector design also translates into energy savings by sending the light only where it is needed. The amount of wasted light for outdoor lighting has been estimated to 30%, which translates into billions of dollars each year [24, 25]. While the advent of LEDs in general lighting will surely help save precious oil, the role of the optics should not be neglected. Poor reflector design leads to blinding glare, obtrusive lighting, and light pollution, whose effects are just starting to be investigated. Light pollution has health consequences on wildlife and humans, most notably circadian rhythm disruption [26, 27]. Poorly designed systems are unfortunately prevalent, as can be seen in aerial pictures of cities in which the glare from distant street light can still be seen from dozens of miles away, as in the example shown in Figure 1.5. Simplifying the reflector design process has the potential to make it more cost-effective and to yield more efficient designs.
As a summary, our research focused on the following goals:

- Implement a reflector design method that takes into account non-Lambertian extended sources and near-field target distributions with or without symmetry.
- Make the design process faster and less reliant on manual trial and error.
- Provide degrees of freedom in the design process for aesthetic considerations.
- Understand the fundamental limits imposed by single reflectors.
- Apply results to concrete general lighting design examples.

### 1.4 Research summary

This section gives a chronological overview of the work that has been accomplished. The work presented here originally started with the study of lightpipe design methods for projection displays [28-31]. Over the years interest shifted towards the design of freeform reflectors, and became the main focus of this dissertation. Early work on lightpipes has been omitted for
conciseness. However, several general design concepts were identified while investigating freeform lightpipes, such as the need to find relevant system parameterizations to make optimization successful and the use of source images as an analysis tool.

Freeform reflectors, even when point sources are used as an approximation, can be challenging to design. We therefore decided to start studying the somewhat simpler case of reflectors with rotational or translational symmetry. The behavior of such reflectors with extended sources was first analyzed, and an iterative method was devised to compensate for the effects of extended sources [32]. Results show that for compact sources, the prescribed target illuminance can be successfully used as an optimization variable to compensate for extended source effects.

Building on this initial work on reflectors with rotational or translational symmetry, we moved on to the design of freeform reflectors. We first implemented the method of supporting ellipsoids developed by Vladimir Oliker, which generates reflectors made of a collection of ellipsoid facets. This method imposes no constraints on the source intensity distribution or the target shape, but it relies on the point source approximation and requires a discrete set of target points. We extended the algorithm so that both crossing and non-crossing geometries could be generated. Various techniques were tested to improve the speed and accuracy of the algorithm, including increment scaling and backward Monte-Carlo ray tracing. We then studied the effects produced by extended sources on the target distribution. The algorithm was adapted so that illuminance of the source images could be directly prescribed, therefore taking into account the source extent during the shape generation process [33]. Finally, we proposed a comprehensive design method with extended sources based on tiling the source images and using scattering to
smooth out the distribution. This method is efficient in cases where slope discontinuities on the reflector are acceptable and no sharp gradients are required in the target distribution.

Since numerous reflector shapes are evaluated during the optimization process, it is critical to make the shape generation algorithm efficient. We investigated ways to generate reflector shapes more efficiently, by directly optimizing the focal parameters of the ellipsoids or by interpolating new ellipsoids from a low-resolution reflector. Still, the current implementation of the algorithm becomes computationally intensive when the number of target points increases. The number of target points we used in practice remained small—typically less than 200.

Freeform reflectors are more complex to address than reflectors with rotational or translational symmetry primarily because there is no generic process to establish mapping relationships between the source and the target when no symmetry exists. Nevertheless, the method of supporting ellipsoids does generate reflector solutions, and hence indirectly provides a source-target mapping. We were able to leverage such source-target mappings to improve the efficiency of the reflector generation process. Mappings can be used to remove slope discontinuities in the original reflector surface and generate smooth surface shapes that produce continuous target distributions. Additionally, because mappings are invariant under small system parameter changes, it is possible to create multiple reflectors with different parameters from the same map, hence avoiding the need to compute a new reflector every time.
1.5 Dissertation outline

Chapter 2 reviews the literature on reflector design. We first review design techniques for systems with rotational or translational symmetry. The point source case is described in detail, since it lays the ground for important design concepts. A quick overview of the methods available for extended Lambertian sources is also given. We then introduce the few recently devised design methods for systems with no symmetry. A section is dedicated to faceted reflectors, which play an important role in general and automotive lighting as they can significantly improve sensitivity to manufacturing tolerances. Finally, we give an overview of the optimization-based techniques that have been used in conjunction with reflector design.

Chapter 3 provides a detailed analysis of the method used to generate freeform reflector shapes. The backbone of our reflector generation technique is the method of supporting ellipsoids developed by Vladimir Oliker. We first give an overview of the geometrical and mathematical principles behind the algorithm. We then detail our implementation of the algorithm, its performance, and possible variations to improve the generation speed. Once the reflector shape has been derived, it must be converted into a 3D shape that can be used in illumination software for analysis. Some details are given on this interpolation process, since it can have a significant impact on the reflector performance.

Chapter 4 analyzes the mapping relationships between source and target. The method of supporting ellipsoids is used to retrieve how source flux is mapped onto the target. We show how the knowledge of these source-target maps can be leveraged to remove slope discontinuities in the reflector, produce continuous target distributions, and make the reflector generation process
more efficient. We also derive the integrability condition under which a mapping can be achieved by a smooth continuous reflector surface. We show that source-target mappings that do not satisfy the integrability condition can be achieved by introducing step discontinuities in the reflector surface, at the expense of artifacts in the target distribution and possible manufacturing challenges.

Chapter 5 focuses on the effects of extended sources on the target distribution. We first quantify the effects of the source extent and detail the limitations imposed by single reflector surfaces. Two methods are then proposed to take into account the source extent during the reflector design process. The first method applies to systems with rotational and translational symmetry, and uses the prescribed target distribution as an optimization variable in order to compensate for the effects of the source extent. The second method applies to systems with no symmetry, and leverages reflectors designed with the method of supporting ellipsoids. The source images produced on the target by such reflectors are tiled, and then blended using scattering off the reflector surface and a diffuser. Both methods are shown to be efficient when the maximum size of the source images is less than about one fifth of the target extent.
CHAPTER TWO - RELATED WORK

This chapter reviews the field of reflector design for illumination, which is often referred to as “tailoring” in the literature. Section 2.1 reviews design methods for reflectors with rotational or translational symmetry. Section 2.2 focuses on design methods for systems without symmetry, referred as freeform reflectors. Section 2.3 gives an overview of the design of faceted reflectors. Finally, Section 2.4 discusses the recent use of optimization in the design process. Whenever possible, the scope and limitations of each tailoring method are indicated. Some papers cited in this chapter specifically deal with tailoring refractive surfaces. Tailoring refractive surfaces adds an additional design constraint: the tailored surface must avoid total internal reflection. Apart from this, tailoring concepts can usually be applied equally for refractive or reflective surfaces, thus unless necessary we will not highlight these differences here.

2.1 Reflectors with rotational or translational symmetry

2.1.1 Point source approximation

When the system has rotational or translational symmetry, the reflector design process becomes a 2D problem in a plane: only a single reflector profile needs to be generated; the full reflector surface can then be obtained by sweeping the curve around its axis of symmetry. Early reflector design relied on graphical methods to construct the reflector profile. Computers have now greatly simplified the process, but the concepts remain the same.
We assume that the total amount of flux collected by the reflector is equal to the required amount of flux on the target (in other words, the light source was chosen appropriately to provide sufficient flux). Our goal is then to shape the source flux into the desired target distribution. In order to do so, we follow a two-step process:

1. Map the flux emitted by the source onto the target
2. Generate the reflector shape that achieves that mapping

First, the designer needs to obtain the angular distribution of the flux emitted by the point source. This angular flux distribution, or intensity distribution $I_s(\theta)$, can be converted into a cumulative flux distribution $\Phi_{\text{source}}$ by integrating the intensity along the emission angle $\theta$ of the source. The geometry for a rotationally symmetric system is shown in Figure 2.1a and the cumulative source flux distribution is given in this case by

$$\Phi_{\text{source}}(\theta) = \int_{\theta_{\text{min}}}^{\theta} I_s(\theta') d\Omega = 2\pi \int_{\theta_{\text{min}}}^{\theta} I_s(\theta') \sin \theta' d\theta',$$  \hspace{1cm} (2.1)$$

where $d\Omega = 2\pi \sin \theta \ d\theta$ is a differential annulus-shaped solid angle, and $\theta_{\text{min}}$ and $\theta_{\text{max}}$ are the limits that define the collection angle of the reflector. The resulting cumulative flux curve is shown in blue in Figure 2.2. This curve indicates how much total flux is contained within a given solid angle. In this example, 40% of the total source flux is emitted within a 42º cone. A similar cumulative flux curve can also be calculated for the desired target distribution (green curve in Figure 2.2). The geometry for a rotationally symmetric near-field target is shown in Figure 2.1b and the cumulative target flux distribution is given in this case by

$$\Phi_{\text{target}}(y) = \int_{y_{\text{min}}}^{y} E_t(y') \, dA = 2\pi \int_{y_{\text{min}}}^{y} E_t(y') \ y' \, dy'.$$  \hspace{1cm} (2.2)$$
Figure 2.1. Geometry of a rotationally symmetric system. (a) Differential annulus-shaped solid angle used when calculating the cumulative source flux. (b) Differential ring-shaped area used when calculating the cumulative target flux for a near-field target.

Figure 2.2. Comparison of the cumulative flux distributions of the source and target in order to establish the relationship between the source ray angles $\theta$ and the target positions $y$. 
By comparing the two cumulative flux distributions, we can obtain a correspondence between the source angles $\theta$ and the target locations $y$. We refer to the function $y = f(\theta)$ as the *mapping* between the source and the target (red curve in Figure 2.2). This mapping process is identical to the one described by Jolley and reproduced earlier in Figure 1.4. In our example, the target is in the near field and hence the target location is specified by the target coordinate $y$. We see in Figure 2.2 that 40% of the flux simultaneously corresponds to $\theta$ equal to $42^\circ$ and $y$ equal to 6.3 cm. This means that a ray emitted by the point source at $\theta = 42^\circ$ should be reflected towards the target location $y = 6.3$ cm. This matching of the cumulative distributions is just a way of applying flux conservation for a specular reflector. If the reflector surface has an average reflectivity $\rho$, then the flux conservation between the source and the target can be written as

$$\rho \Phi_{\text{source}} (\theta) = \Phi_{\text{target}} (y)$$

(2.3)

where $\rho$ is the average reflectivity of the reflector, $I_s(\theta)$ and $E_t(y)$ are respectively the source intensity and the prescribed target illuminance. This flux conservation equation takes different forms depending on the geometry of the problem. For instance, for a far-field target and a rotationally symmetric system, one needs to calculate the target flux contained within an annular-shaped differential solid angle. In this case, we thus have

$$\rho \int_{\theta_{\text{min}}}^{\theta} I_s (\theta') \sin \theta' d\theta' = \int_{y_{\text{min}}}^{y} E_t (y') y' dy',$$

where $\rho$ is the average reflectivity of the reflector, $I_s(\theta)$ and $E_t(y)$ are respectively the source intensity and the prescribed target illuminance. The flux conservation equations for each configuration are summarized in Table 2.1 (the sign before the integral depends on how the integration bounds are chosen).
Table 2.1. Flux conservation equations for various reflector geometries.

<table>
<thead>
<tr>
<th>Symmetry</th>
<th>Far-field target</th>
<th>Near-field target</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rotational</td>
<td>$\rho \int I_s(\theta) \sin \theta d\theta = \pm \int I_r(\beta) \sin \beta d\beta$</td>
<td>$\rho \int I_s(\theta) \sin \theta d\theta = \pm \int E_i(y) , y , dy$</td>
</tr>
<tr>
<td>Translational</td>
<td>$\rho \int I_s(\theta) d\theta = \pm \int I_r(\beta) , d\beta$</td>
<td>$\rho \int I_s(\theta) d\theta = \pm \int E_i(y) , dy$</td>
</tr>
</tbody>
</table>

Figure 2.3. The reflector polar radius $r$, and the angles $\theta$ and $\beta$ of the ray before and after reflection are linked through the traditional reflector equation (Equation 2.2).

Once the source-target mapping is known, the reflector shape can be constructed iteratively by ensuring that the surface normal at each newly constructed point reflects the rays towards their corresponding target location. As we can see in Figure 2.3, the relationship between the angles $\theta$ and $\beta$ and the reflector radius in polar coordinates is given by

$$\frac{dr}{r} = \tan\left(\frac{\theta - \beta}{2}\right) d\theta,$$

(2.5)
where \( r \) and \( \theta \) are respectively the polar radius and the polar angle of the reflector, and \( \beta \) is the direction of the ray after reflection for each angle \( \theta \). The reflector radius as a function of the angle \( \theta \) is the solution of a differential equation, which can be put in integral form as

\[
r(\theta) = r_{\text{min}} \exp \left[ \int_{\theta_{\text{min}}}^{\theta} \tan \left( \frac{\theta - \beta}{2} \right) d\theta \right], \tag{2.6}
\]

where \( \theta_{\text{min}} \) is the polar angle at the inner rim of the reflector, and \( r_{\text{min}} \) is the polar radius at \( \theta_{\text{min}} \) (the apex if \( \theta_{\text{min}} = 0 \)). \( r_{\text{min}} \) acts as a scaling parameter which controls the overall size of the reflector. Alternatively, integration can be started at \( \theta_{\text{max}} \) so that the polar radius \( r_{\text{max}} \) at the outer rim of the reflector can be set. This scaling is irrelevant for a theoretical point source, but it will become critical when we consider real extended sources. If the angle \( \beta \) can be expressed as a function of \( \theta \), then Equation 2.6 can be solved analytically. Otherwise, Equation 2.3 can be numerically integrated using the derived mapping between \( \theta \) and \( \beta \).

When the target is located in the near field, \( \beta \) must be expressed as a function of the target coordinates \((y, z)\) and the polar radius \( r \) of the reflector. Using Figure 2.4, we find

\[
\beta = \arctan \left( \frac{r \sin \theta - y}{z + r \cos \theta} \right). \tag{2.7}
\]

The differential equation governing the reflector shape now becomes

\[
\frac{dr}{d\theta} = r \tan \left[ \frac{\theta}{2} - \frac{1}{2} \arctan \left( \frac{r \sin \theta - y}{z + r \cos \theta} \right) \right]. \tag{2.8}
\]

For a flat target perpendicular to the axis, \( z \) is a constant corresponding to the target location, and \( y \) can be expressed as a function of \( \theta \) using the derived source-target mapping. This type of non-linear differential equation can easily be solved with standard numerical integration techniques such as Runge-Kutta methods.
The reflector equations for the rotationally symmetric case seem to appear in the 1930s. In fact, equations were derived at the same time for the general case with no symmetry by Boldyrev [12], Komissarov [34], and later on by Schruben [35], but until recently no numerical solutions had been found, so rotational and translational symmetry remained the cornerstone of most design techniques. Thorough treatments of reflector design within the point source approximation can be found in Elmer [15] and in Kush [36].

The reflector shape corresponding to a given prescribed target distribution is not unique. Multiple source-target mappings producing the same result are possible. Figure 2.5 shows the four possible solutions, which we refer to as crossed diverging, uncrossed diverging, crossed converging and uncrossed converging (the terminology varies in the literature). Note that these are the only four possible solutions when a one-to-one mapping is achieved between the source and the target, i.e. when every point of the reflector is mapped to a unique point on the target. On the contrary, faceted reflectors are made of multiple facets that create overlapping beams, so the
same point on the target receives light from multiple regions of the reflector. Section 2.4 will provide more details on faceted reflectors. These four mappings yield reflectors of very different shapes and sizes, as shown in Figure 2.5. Uncrossed reflectors minimize interference of the reflected light with the source but they can become impractically large. The crossed diverging solution yields a compact reflector but most of the reflected light interferes with the lamp, so this configuration is rarely used in practice. The crossed converging solution is a good trade-off, with a compact shape and minimal lamp interference. However, the caustic formed by the rays can be undesirable in some cases: any object placed close to the caustic would have a tremendous effect on the output distribution. For these reasons, hybrid geometries combining multiple reflector sections with different sections are often used.
Figure 2.5. Reflector shapes for the four possible source-target mappings. While all mappings produce identical target distributions, they lead to very different reflector geometries.
2.1.2 Extended sources

The range of validity of the point source approximation is difficult to establish in general. The important criterion is the apparent size of the source as seen from the reflector. Each point on the reflector can be thought of as creating its own image of the source on the target. The points $A$, $B$ and $C$ in Figure 2.6 create pinhole source images with various magnifications depending on the apparent size of the source as seen from each point. In this example, the source image at the rim of the reflector (blue ray fan) is much smaller than at the apex (red ray fan), since the rim is located further away from the source. As the reflector size decreases, the images of the source on the target scale up, and a single point on the reflector may have an impact on a large portion of the target. This is the reason why compact reflectors are challenging to design. The typical effects of an extended source on the target distribution are smearing of the target edges and peaks or dips at its center. These effects are demonstrated in Table 2.2, which shows the evolution of an illuminance distribution as the size of a cylindrical source gradually increases.

Figure 2.6. Effect of an extended source on the target distribution. The relative size of the source as seen from the reflector determines its impact on the target distribution. The closer the reflector is to the source, the larger the corresponding source image will be.
Table 2.2. Effect of a cylindrical source on the target distribution as its size increases.

<table>
<thead>
<tr>
<th>Reflector geometry</th>
<th>3D illuminance plot</th>
<th>Cross-section</th>
</tr>
</thead>
<tbody>
<tr>
<td>Point source</td>
<td><img src="image1" alt="Diagram" /></td>
<td><img src="image2" alt="Diagram" /></td>
</tr>
<tr>
<td>Cylindrical source, ( R = 1.5)mm</td>
<td><img src="image3" alt="Diagram" /></td>
<td><img src="image4" alt="Diagram" /></td>
</tr>
<tr>
<td>Cylindrical source, ( R = 3)mm</td>
<td><img src="image5" alt="Diagram" /></td>
<td><img src="image6" alt="Diagram" /></td>
</tr>
<tr>
<td>Cylindrical source, ( R = 6)mm</td>
<td><img src="image7" alt="Diagram" /></td>
<td><img src="image8" alt="Diagram" /></td>
</tr>
</tbody>
</table>
Elmer gives as a criterion that when the source images extend beyond 5% of the target size, corrections will be necessary to account for the source extent [15]. He describes two techniques to cope with undesirable effects of extended sources. First, the “Phantom light center” is useful for removing excess brightness at the center of the beam. Instead of placing the point source at the geometrical center of the extended source, it is placed outside the source, so that rays appear to originate from all over the extended source surface as we move along the reflector profile. The second technique described by Elmer, extinction curves (sometimes also referred to as macrofocal curves [37]), are a mean to obtain a sharp cutoff with an extended source, by forcing all rays coming from the edge of the source not to be reflected beyond the edge of the target. Bortz proposed a similar approach where the point source can move along the source surface as the reflector profile is being generated [38]. With a single reflector and an extended source, only a partial control of the output distribution can be obtained. For instance, requiring a sharp cutoff dictates a significant portion of the reflector profile and does not leave enough degrees of freedom to fully control the rest of the distribution. Conversely, controlling the center of the distribution tightly leaves no control over the edge falloff.

A new wave of interest for tailored reflectors with extended sources started in the early 1990s, when Winston suggested that traditional solar concentrator design techniques could be leveraged for illumination [39, 40]. This gave birth to a prolific series of papers about tailored edge-ray designs (TED). If a Lambertian source illuminates a reflector, then by conservation of luminance the specular reflector becomes a secondary Lambertian source. The intensity at a given point on the target can be derived from the flashed area of the reflector. The flashed area corresponds to
the region of the reflector shining light in a given target direction (see Appendix B for more
details). By using rays from the edges of the source, we can tailor the reflector shape in order to
control the flashed area, and thus the intensity, at each point on the target.

The general TED technique was introduced by Winston for a Lambertian strip source [41]. As
with a point source, there are four possible mappings and thus four possible reflector shapes that
achieve the same result. These four solutions are detailed by Rabl and Gordon for strip sources
[42]. The results were extended to tubular sources by Ong [43]. A solution with no gap between
the reflector and the source was proposed by Jenkins [17]. Gordon explored the intrinsic
limitations of rotationally symmetric devices [44] and ultimately reached the same conclusions
as Elmer [15]: when a uniform distribution is required at the center of the target, no degrees of
freedom are left to control the edge falloff. All the TED methods are explained in great details by
Chaves [45].

These TED methods are devised in 2D, and by extension they apply to systems with translational
symmetry. However, they cannot be directly extended to 3D: for instance, it is not possible to
obtain a rotationally symmetric concentrator by simply revolving the reflector shape obtained in
2D. Several solutions have been proposed to overcome this limitation for disc sources, by using
iterative methods [44, 46] or a compensated 2D process [47]. Finally, Korobko and Kush
developed a method that handles cylindrical sources [48].

All the previously mentioned design techniques in this section and the corresponding references
are summarized in Table 2.3. TED methods are all limited by the underlying assumption that the
source is Lambertian and were derived for far field targets only.
<table>
<thead>
<tr>
<th>Geometry</th>
<th>Source</th>
<th>Symmetry</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>Point</td>
<td>Rotational</td>
<td>Early graphical</td>
<td>Early graphical methods, Jolley [10]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Detailed near-field case, Kusch [36]</td>
</tr>
<tr>
<td>Line</td>
<td>Translational</td>
<td>Same references as</td>
<td>Same references as point source case</td>
</tr>
<tr>
<td></td>
<td>symmetry</td>
<td>point source case</td>
<td></td>
</tr>
<tr>
<td>Strip, Lambertian</td>
<td>Translational</td>
<td>General concept,</td>
<td>General concept, Winston [41]</td>
</tr>
<tr>
<td></td>
<td>symmetry</td>
<td>Winston [41]</td>
<td>General solutions, Rabl [42]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Detailed review, Chaves [45]</td>
</tr>
<tr>
<td>Tube, Lambertian</td>
<td>Translational</td>
<td>General concept,</td>
<td>General concept, Ries [16]</td>
</tr>
<tr>
<td></td>
<td>symmetry</td>
<td>Ries [16]</td>
<td>General solutions, Ong [43]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Solution with no gap, Jenkins [17]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Detailed review, Chaves [45]</td>
</tr>
<tr>
<td>Disc, Lambertian</td>
<td>Rotational</td>
<td>Iterative method,</td>
<td>Iterative method, Gordon [44]</td>
</tr>
<tr>
<td></td>
<td>symmetry</td>
<td>Gordon [44]</td>
<td>Iterative method, Benítez [46]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Compensated 2D process, Yang [47]</td>
</tr>
<tr>
<td>Cylinder,</td>
<td>Rotational</td>
<td>Zonal method,</td>
<td>Zonal method, Korobko and Kusch [48]</td>
</tr>
<tr>
<td>Lambertian</td>
<td>symmetry</td>
<td>Korobko and Kusch [48]</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Kusch [36]</td>
<td></td>
</tr>
</tbody>
</table>
2.2 Freeform reflectors

2.2.1 Equi-flux grid methods

When the system has neither rotational nor translational symmetry, the design problem becomes three-dimensional. Some reflected rays do not remain in the meridional plane of the system any more. For rotationally or translationally symmetric systems, we could derive the mapping between the source and the target by comparing their cumulative flux distributions, because in that case the cumulative flux is a simple 1D integral. Without symmetry, the cumulative flux must be computed over a two-dimensional space. We now have

$$\iiint I(u,v)\,du\,dv = \iiint E(u',v')\,du'\,dv',$$

where $I(u,v)$ is the intensity distribution of the source and $E(u',v')$ the desired illuminance distribution over the target. In practice, we often use spherical coordinates for the source and Cartesian coordinates for a near-field target, so the previous expression becomes

$$\iiint I(\theta,\phi)\sin \theta \,d\theta\,d\phi = \iiint E(x,y)\,dx\,dy.$$

Finding a mapping between the source and the target means that we have to express $x$ and $y$ as functions of $\theta$ and $\phi$ so that

$$x = f (\theta,\phi),$$
$$y = g (\theta,\phi).$$

Unfortunately, there is no general way to find a mapping unless some common symmetry exists between the source and the target, in which case we can separate the integration variables and obtain the dependency on $\theta$ and $\phi$ independently. Figure 2.7 shows an example of an isotropic point source and a uniform square target. Both the source and the target distributions are divided
into bins that contain the same amount of flux. Since the source is isotropic and the target uniform, each source bin has the same solid angle and each target bin has the same area. The mapping is obtained by matching the two grids, as we shall see in more details in Chapter 4. In this case we leveraged the radial symmetry of the system. Many other common target shapes exhibit similar radial symmetry, such as regular polygons, ellipses, stars, and crosses.

Figure 2.7. Equi-flux grids used to derive a mapping between the source and the target.

If we can establish an analytical expression for the mapping, we can plug it into the reflector equation, which can then be numerically integrated to derive the reflector shape. This approach was reported by Ding [49]. Since numerical integration of these equations can still be cumbersome, a simple geometrical construction can be used as an alternative way to generate the reflector shape [50, 51]. In these cases, the mapping equations are converted into a discrete grid
of source ray directions and a corresponding grid of target points. The surface is then constructed iteratively so that each source ray reaches its corresponding target point.

The major problem with all grid-based techniques is that an arbitrary ray mapping does not ensure that the generated surface will be smooth and continuous. Step discontinuities might be unavoidable in some cases. If we force continuity during surface construction for a given mapping, we will typically create slope errors that in turn generate errors in the target distribution. Finding a mapping is therefore not sufficient; we need to find a mapping that ensures surface continuity. As we will see in Chapter 4, only mappings that fulfill the integrability condition can be achieved by smooth reflector surfaces. When the integrability condition is not fulfilled, we can either introduce step discontinuities in the surface, as explored by Parkyn [52], Wang [51] and Ding [49], or use methods that do not rely on a priori knowledge of the source-target mapping, as we shall see next.

2.2.2 3D tailoring methods

In general, real sources do not produce symmetric emission patterns, so there is no easy way to find a source-target mapping. Even if there were, there is no guarantee that a given mapping would satisfy the integrability condition, which is necessary to achieve a continuous reflector surface. For systems with no symmetry, mapping and surface generation cannot be conveniently separated. Tailoring methods do not make a distinction between mapping and shape generation: they treat the problem as a whole. The reflector shape is derived from a system of three equations: flux conservation, law of reflection, and surface integrability, as shown in Table 2.4.
A boundary condition is also required to scale the reflector. Numerous formulations of this problem have been given over the years [12, 34, 35]. They all demonstrate that the system of equations governing the shape is equivalent to a non-linear second-order partial differential equation of the Monge-Ampère kind. This type of equation can quickly become unstable when solved with standard numerical integration techniques. Nonetheless, two solutions using different approaches have been recently proposed for the near-field case by Oliker and by Ries.

Table 2.4. System of equations governing the general reflector problem

<table>
<thead>
<tr>
<th>Equation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flux conservation</td>
<td>( \iint_S I(r_{\text{in}}) , d\Omega = \iint_T E(t) , dA ) (2.12)</td>
</tr>
<tr>
<td>Law of reflection</td>
<td>( N = \frac{r_{\text{out}} - r_{\text{in}}}{\sqrt{2(1 - r_{\text{out}} \cdot r_{\text{in}})}} ) (2.13)</td>
</tr>
<tr>
<td>Integrability condition</td>
<td>( N \cdot (\nabla \times N) = 0 ) (2.14)</td>
</tr>
</tbody>
</table>

Oliker proposed an iterative method where the reflector surface is made of a combination of ellipsoids [19]. This method will be explored at great length in Chapter 3. The key aspect of this method is that it uses a discrete target. Each ellipsoid constituting the reflector reflects light towards a specific target point. The iterative process then ensures that the correct amount of flux is collected by each ellipsoid and thus reflected to each target point. While this method is conceptually intuitive, there is no easy way to convert the discrete distribution produced by the generated reflector into a smooth, continuous distribution. On the other hand, increasing the
number of target points greatly increases the required computational time to converge to a solution.

Ries and Muschaweck outlined a formulation of the problem based on wavefronts that can be solved numerically using multigrid methods [20]. However, few details are given on the actual numerical solution procedure, which is the key to this problem. While these results are a breakthrough, they have not been reproduced yet, most likely because the method has been kept proprietary.

Regardless of the design method, it is worth noting that the contour of the reflector cannot be arbitrarily specified. This would in general violate the integrability condition, since there is no guarantee that path integration along the boundary would yield the specified contour. Instead, the contour can be constrained to lie on a cone or a cylinder. Some design examples for general illumination are presented in [53]. As in the 2D case, there are usually four possible topological mappings, and hence four distinct reflector shapes, that all achieve the same output distribution.

For the far field case, the problem is somewhat simpler, since the ray angle after reflection is independent of the location where the ray hits the reflector (simpler, not simple!). This problem has been investigated in the field of antenna design, so there is abundant literature on the topic. Wang provides a good review of the mathematical advances in the field [54], and proposes a solution to the reflector problem as the maximization of a linear functional subject to a linear constraint [55]. Oliker’s method of supporting ellipsoids can simply be extended to the far-field case by using paraboloids instead of ellipsoids.
2.2.3 Multiple surfaces

Multiple lenses are often necessary to obtain good image quality. Similarly, a single surface does not always have enough degrees of freedom to obtain the required illumination or intensity pattern. By “single surface”, we include surfaces with possible slope and step discontinuities. The only constraint is that each ray interacts with the surface only once. A single surface, in that sense, cannot fully control the size and orientation of the source images. To understand this, let’s consider an elementary surface patch. Once we define where an incoming ray must hit the target, the surface normal of the patch can be set according to the law of reflection. There is no degree of freedom left to choose the reflected direction of a second ray hitting the same reflector patch (a second surface would be required to do so), so the orientation and the size of the source image is set as well.

As a consequence, it is not possible to have complete control over the output distribution. One way to overcome these limitations is to add additional surfaces. An elegant method to construct two surfaces simultaneously is the SMS (Simultaneous Multiple Surfaces) method developed by Miñano and Benítez [18]. Using two surfaces means that two points can be perfectly imaged to two points (or more generally, two incoming wavefronts can be arbitrarily transformed). This approach is well suited for the design of concentrators based on the edge-ray theorem [14]. On the other hand, using the SMS method for tailoring applications involves a complex choice of input and output wavefronts. A design example for a LED headlight is given in [56], and a condenser design using two mirrors was presented in [57].
2.3 Faceted reflectors

The reflectors we have been discussing so far achieve a one-to-one mapping between the source and the target. Each region of the target is illuminated by a unique region of the reflector. This approach makes the target distribution sensitive to the reflector shape and the source characteristics. If the source position shifts, the source emission pattern changes slightly, or the surface shape is slightly altered, then the target distribution is directly impacted. In order to reduce the sensitivity of the system, the reflector can be divided into multiple facets that all illuminate the same target region. This way, small local variations in the reflector shape or the source emission pattern will have a small overall impact on the target distribution. This approach is very practical for uniform target distributions.

With a point source, the facet shape governs the shape of the output distribution. The reflector shown in Figure 2.8 uses flat rectangular-shaped facets. In this case, facets are tailored in order to produce overlapping rectangular patterns on the target. When the pattern is revolved around the axis, then the superposition of rectangular patterns on the target yields a uniform disc with a gradual falloff region, as shown in Figure 2.8. With an extended source, the pattern generated by each facet is the result of the convolution between the “facet response” and the source image produced by each facet [58].

An attractive feature of faceted reflectors is their design flexibility. Many different designs with different facet shapes and sizes yield similar output distributions. The choice of a specific design is often guided by aesthetic considerations, which play a critical role among other design
requirements. A more detailed study of facet generation algorithms can be found in [58]. A thorough review of the use of faceted structures in illumination is given in [59].

Figure 2.8. Uniform illumination with a faceted reflector. Facets along each wedge are tailored in order to illuminate the same region of the target. The superposition of similar patterns with different orientations produces uniform illuminance in the center and gradual edge falloff.
All facets do not have to produce identical output patterns. Faceted reflectors in which facets produce distinct patterns that only partially overlap are common in automotive lighting. Facets are used as building blocks to create the complex output distributions required for headlamps. An example of such reflector taken from a patent dating from 1987 is given in Figure 2.9. The drawing on the right shows how each section of the reflector is tailored for a specific region of the output beam. Two types of facets are commonly used: parabolic shapes create bright spots by focusing rays in one direction, while cylindrical shapes widen the beam along one dimension. Design techniques for headlamp are often proprietary, but a couple of papers give a good overview of the design process [60, 61].

![Faceted reflector used in an automotive headlamp (US Patent No 4,704,661).](image)

The typical design process for a faceted reflector headlamp is as follows:

- Define the reflector aperture shape and the base surface along which facets are placed.
- Dissect the target and the reflector into multiple zones according to the desired target distribution and the average flux collected by each facet.
- Tailor each facet individually. This is often achieved by adjusting the tilt, curvature and cylinder of each facet. Alternatively, any tailoring method described in this chapter could be used.
- Merge facets together by filling gaps and eliminating shadowed regions.
- Evaluate the overall target distribution and adjust the shape iteratively.

Headlamps are another example of a design process being guided predominantly by aesthetics. Nowadays, headlights form factors must be adapted in order to match the shape of the car body, which adds considerable design constraints (but since each new car model requires a new design, this is good news for reflector designers). The outside lens typically acts as a cover only, while the reflector achieves most of the beam shaping. When LEDs are used instead of HID lamps, refractive designs and TIR lenses can become advantageous.

Altogether, faceted reflectors provide robustness and cover a wide range of applications. Faceted reflectors with full overlap are well suited for uniform distributions with rotational symmetry. On the other hand, reflectors with partial overlap are useful for more complex patterns, but their design remain extremely time-consuming. One of the goals of our work is to simplify this process by embedding optimization. The next section reviews prior work in this area.

2.4 **Optimization techniques for reflector design**

Two parameters are essential when setting up an optimization problem: the merit function and the choice of optimization variables. The choice of a merit function is somewhat limited for reflector design: it can either be the illuminance measured at a target point or the position of
specific ray intercepts on the target. As for the optimization variables, two strategies prevail: using the shape descriptors as variables, such as polynomial coefficients; or using target descriptors as variables, such as the prescribed illuminance distribution. In the latter case, an algorithm then constructs the reflector shape based on the prescribed distribution (since the algorithm usually makes simplifying assumptions about the source, optimizing the prescribed distribution is a way to compensate for the source departure from ideality). This way, complex shapes can be generated with fewer variables. In all cases, the number of variables and the number of rays traced during target evaluations must be kept small in order to keep optimization time under control.

Early attempts at optimization for reflector design strongly depended on the ability to evaluate target distributions with very few rays, given the minimal computational power and memory available at the time. Single reflectors, because of their simple geometry, allow a couple of efficient solutions, such as flux tubes and backward ray tracing. As early as 1972, Vogl described two optimization methods he refers as “field patch mode” and “aperture flash mode” [62]. These two setups are depicted in Figure 2.10a. In field patch mode, a grid of pinholes is placed in front of the reflector and the corresponding source image produced on the target by the pinholes is evaluated with a small number of rays. The target illuminance is approximated by the superposition of all the source images. Optimization can be used to shuffle around the source images in order to match the desired illuminance, and corresponding changes can then be made to the shape (no details are given on this process, however). Donohue proposed the same year a similar optimization technique for faceted reflectors [60]. In aperture flash mode, on the other
hand, illuminance at a set of target points is calculated from a quick evaluation of the flashed area of the aperture, as shown in Figure 2.10b. The reflector shape is then optimized in order to minimize the deviation between the measured values and the prescribed values at these points.

![Figure 2.10](image)

Figure 2.10. Optimization techniques described by Vogl in 1972 (reproduced from [62]). (Left) In field patch mode, a grid of pinholes is placed in front of the reflector and the corresponding source images are used for analysis. (Right) In aperture flash mode, illuminance at a set of target points is evaluated by estimating the flashed area of the aperture at each point.

In 1995 Rabl integrated a tailored edge-ray design into an iterative algorithm as a mean to handle non-isotropic sources [63]. A correction factor was applied to the prescribed distribution based on the deviation between measured and prescribed illuminance values. The adjusted prescribed values are used to generate a new reflector shape, and so on, until an acceptable result is obtained. However, the examples presented show slow convergence and very limited improvement after several iterations. Bortz and Shatz proposed a similar iterative approach for reflectors with rotational symmetry based on the point source approximation [64].

Davenport studied the direct optimization of NURBS patches for uniform circular illumination [65, 66], as well as the optimization of XY polynomial surfaces for uniform square illumination.
Surface parameters such as NURBS control points or XY polynomial coefficients are used as optimization variables. The merit function is usually either a grid of rays aimed at specific target locations, or triads of rays used to quickly evaluate the illuminance at the target (flux tubes). Good results were obtained for small collection angles and a 9×9 grid of NURBS control points (hence 24 optimization variables if we factor in quadrant symmetry).

Sikkens explored semi-automatic optimization of polynomial shapes for faceted reflectors used in automotive fog lamps [68]. The optimization process follows three steps. The user first defines the reflector segments and their corresponding regions on the target and adjusts manually the base shape of each facet in order to get a coarse approximation of the required distribution (position, tilt, base curvature). The shape of each facet is then optimized by using polynomial coefficients as optimization variables. As the merit function decreases, the number of traced rays and the resolution of the target are gradually increased. This helps preventing optimization algorithms from getting stuck in local minima. It is worth noting that as of 2010, software packages dedicated for reflector designs, such as LucidShape® and ReflectorCAD®, do not use optimization. Instead, they streamline the traditional trial-and-error process by letting the designer modify interactively the surface shape and get direct feedback. This, however, can still be time-consuming and require significant expertise.

Other attempts at a direct optimization of the shape include [69], where the reflector shape is described by a mesh. The location of the mesh vertices is optimized within a range defined by the user, and the number of surface vertices is gradually increased as the merit function
improves. Incidentally, this approach highlights the need for a good starting point and the considerable computational power required when the number of variables is high.

Finally, Michaelis explored the use of optimization in conjunction with the method of supporting paraboloids developed by Oliker [70]. The next chapter gives more insight on this method and how to use it to generate freeform reflectors.
CHAPTER THREE - SHAPE GENERATION

This chapter covers reflector generation methods for systems without rotational or translational symmetry. Sections 3.1 to 3.5 deal with the method of supporting ellipsoids developed by Oliker, which can be used both for far-field and near-field targets. Section 3.1 introduces the concepts behind the method of supporting ellipsoids. Section 3.2 focuses on practical implementation details and proposes two shape generation algorithms. Section 3.3 deals with methods to evaluate the target distribution. Section 3.4 covers the conversion of the generated shape made of ellipsoid facets into a freeform surface that can be exported to other software for analysis. Section 3.5 provides an “atlas” of generated reflector shapes for typical system configurations. Finally, Section 3.6 presents an alternative shape generation method based on a variational formulation of the problem for far-field targets only.

3.1 Overview of the method of supporting ellipsoids

The backbone of the shape reflector generation method we are using is the method of supporting ellipsoids developed by Oliker. For our purpose, we will present the concepts behind the algorithm in an intuitive manner. More detailed information about the mathematical framework can be found in [19].

Let’s consider a solid angle over which we want to collect the light emitted by a point source. This collected light is “mapped” by a reflector to a target according to the desired illuminance
distribution. In order to solve this problem numerically, we will consider a discrete target with a finite number of points (we can think of the target as if it were made of pixels). Each target point is assigned a prescribed value, as shown in Figure 3.1. If the desired target distribution is uniform, then uniformly spaced target points will be assigned the same value. The total flux received on the target is always normalized so that it matches the flux emitted by the source and collected by the reflector. The absolute target values have thus no importance; only the relative differences between target values are relevant to describe the target illuminance distribution.

The next step is to find a shape that reflects light to each target point with its correct relative proportion. Ellipsoids perfectly reflect light emitted from one of their foci to their other focus, as illustrated in Figure 3.2a. In our case, we place one focus of the ellipsoid at the point source location and the other focus at a given target point location. That way, we know that any light emitted from the source that hits an ellipsoid will be reflected to its corresponding target point. We therefore need as many ellipsoids as there are target points. All ellipsoids share the point source as common focus.
Figure 3.2. (a) Ellipsoids reflect all the rays emitted from one of their foci to the second focus. (b) Ellipsoid parameterization.

Using Oliker’s notation, the equation of an ellipsoid in polar coordinates is given by

$$\rho(\mathbf{m}) = \frac{d}{1 - e \mathbf{m} \cdot \hat{\mathbf{v}}} \quad \text{with} \quad e = \sqrt{1 + \frac{d^2}{|\mathbf{v}|^2} - \frac{d}{|\mathbf{v}|}}, \quad (3.1)$$

where $\rho$ is the polar radius of the ellipsoid from $F_1$, $d$ is the focal parameter, $e$ is the eccentricity, $\mathbf{v} = \overrightarrow{F_1F_2}$ is the vector joining the two foci, $\hat{\mathbf{v}} = \mathbf{v}/|\mathbf{v}|$ and $\mathbf{m}$ is a unit vector corresponding to the incident direction, according to Figure 3.2b. For an ellipsoid, $0 < e < 1$. Once the two foci are specified for each ellipsoid, the only degree of freedom left to entirely define the ellipsoid shape is the focal parameter $d$. Figure 3.3 shows a family of ellipsoids that all have coincident foci (which means that for any of these ellipsoids, light emitted from $F_1$ will be reflected to $F_2$). The only difference is their focal parameter $d$. Alternatively, we can rewrite the ellipsoid equation from Eq. 3.1 using another commonly used parameterization

$$\rho(\mathbf{m}) = \frac{a^2 - c^2}{a - c \mathbf{m} \cdot \hat{\mathbf{v}}}, \quad (3.2)$$

where $a$ is the semi-major axis and $2c$ is the distance between the two foci, as shown in Figure 3.2b. In this case $a$ acts as the scaling parameter instead of the focal parameter $d$. 

69
Figure 3.3. Family of confocal ellipsoids. All ellipsoids have the same foci $F_1$ and $F_2$ but different focal parameters. The focal parameter acts as a scaling factor for the ellipsoids.

More details on the various ellipsoid parameterizations can be found in Appendix C. The two parameterizations in Eq. 3.1 and Eq. 3.2 are linked by the relationships

$$2a = d + \sqrt{|v|^2 + d^2} \quad \text{and} \quad 2c = |v|.$$  \hspace{1cm} (3.3)

In Figure 3.4, we show a simple example with two ellipsoids. Scaling lets us define how much flux is intercepted by each ellipsoid, and therefore how much flux is collected at each target point. In order to do so, we must define a rule that defines which ellipsoids reflect which rays.

Figure 3.4. Definition of ellipsoids intersection rules. A rule is defined to determine how to generate the reflector surface from the combination of all ellipsoids. In this example, the first intersected ellipsoid reflects the ray to its corresponding target point. The ray on the left will therefore be reflected by $E_i$ at $I_i$ and hit target point $T_i$.  

70
One possible rule is: “the first ellipsoid intersected by a given ray reflects that ray to its corresponding target point”. The ray emitted from the source respectively intersects the two ellipsoids $E_1$ and $E_2$ at $I_1$ and $I_2$. According to the rule we defined, $E_1$ is the first ellipsoid to be intersected by the ray. Therefore, the ray will be reflected by $E_1$ and hit target point $T_1$, which is the second focus of $E_1$. In this case, the final reflector surface corresponds to the set of ellipsoid pieces that lie the closest to the source, as shown in Figure 3.5. Alternatively, we can obtain a reflector made of the set of ellipsoid pieces that lie the farthest away from the source by using the last interested ellipsoid, as shown on Figure 3.6. The choice of rule therefore produces two different reflector geometries, crossing or non-crossing, which both produce the same target distribution. This family of solutions is similar to the rotationally symmetric case, except that in this case we only have access to two reflector shapes instead of the four shapes shown in Figure 2.5.

Once we have chosen a rule, we need to find the focal parameter $d_i$ for each ellipsoid that will yield the correct flux at each target point. This is the core of the numerical solution. We present two different approaches to scale the ellipsoids. The original method presented by Oliker uses an iterative algorithm that is guaranteed to converge to a solution [71]. An alternative approach, investigated in [70], uses direct optimization of the focal parameters to yield a solution. In this case, convergence to a solution is contingent on the setup of the optimization problem. Both solutions are detailed in Section 3.2. In all cases one ellipsoid must remain fixed: this “reference ellipsoid” sets the overall scale of the reflector and the uniqueness of the solution. For instance, by using ellipsoid $E_2$ (corresponding to an on-axis target point) as the reference ellipsoid in Figure 3.6, one can set the distance between the source and the apex of the reflector. During the
scaling process all other ellipsoids will then scale relatively to $E_2$. On the other hand, if the distance from the source to the rim of the reflector is fixed, one can use an ellipsoid at the edge of the reflector such as $E_1$ or $E_3$ as the reference ellipsoid.

Figure 3.5. Crossing reflector geometries are obtained when using the set of ellipsoid facets the closest to the source.

Figure 3.6. Non-crossing reflector geometries are obtained when using the set of ellipsoid facets the farthest away from the source.

Figure 3.7 shows a 3D view of crossing and non-crossing reflectors obtained using the method of supporting ellipsoids. Each reflector is made of 25 patches of ellipsoids. The target is a 5x5 grid of equally spaced points with identical target values (uniform square illumination).
So far we have considered flat targets located in the near-field. This is by no means a limitation of the method. The target may have any shape, the source any intensity distribution, and the reflector any arbitrary collection angle. When the target is located in the far-field and thus defined as an intensity distribution, ellipsoids degenerate into paraboloids and the same procedure can be used. Also, refractive surfaces can be created instead of reflective surfaces by using Cartesian ovals instead of ellipsoids. In this case special care must be taken to deal with possible total internal reflection within the refractive material.
3.2 Algorithm implementation

3.2.1 Algorithm A: Modified Kochengrin algorithm

The original algorithm proposed by Kochengrin and Oliker [72] generates a sequence of reflectors that converges monotonically to the desired solution. To illustrate the method we first consider the simple case with two ellipsoids shown in Figure 3.8.

![Diagram showing ellipsoid scaling process](image)

Figure 3.8. Ellipsoid scaling process. By decreasing the focal parameter $d_1$, ellipsoid $E_1$ gradually transitions from collecting 0% of the flux to collecting 100% of the flux. One value of the parameter $d_1$ thus yields the desired amount of flux at target point $T_1$.

$E_0$ is chosen as the reference ellipsoid and hence remains fixed. Initially, all other ellipsoids are scaled so that they are larger than $E_0$. According to the “closest ellipsoid” intersection rule, all the flux emitted by the source reaches $T_0$. We now start decreasing the focal parameter $d_1$ of $E_1$. At some point, $E_1$ will become smaller than $E_0$, hence all rays will be reflected to $T_1$. Since this is a continuous problem, the flux collected at $T_1$ can take all values from 0% of the source flux to 100% of the source flux as we decrease $d_1$. There is therefore a value of $d_1$ for which we obtain
the desired target flux at $T_i$. If more than two ellipsoids are used, we repeat the same process for $E_2$, and so on, until we reach the last ellipsoid.

Figure 3.9. Flowchart of the algorithm based on the method of supporting ellipsoids.

Scaling down $E_i$ has an effect on all previously scaled ellipsoids. However, by repeating the entire scaling process over, the reflector ultimately converges to a solution that achieves the
desired flux at each target point. The algorithm steps are summarized in the flowchart in Figure 3.9. We discuss next the implementation of three important steps in the algorithm: setting the reflector size, choosing the initial focal parameters, and scaling iteratively the ellipsoids.

**Setting the reflector size**

The choice of the focal parameter $d_0$ of the reference ellipsoid can be used to control the overall size of the reflector. The size of a reflector is typically defined by either setting the distance between the source and the reflector apex, or by setting the distance between the source and the outer rim of the reflector, as shown in Figure 3.10. Since freeform reflectors do not in general have a constant distance between the source and the rim (especially when the target is strongly asymmetric), we need to be a little more specific and indicate in which direction we wish to set the source–rim distance. The key to setting the reflector size is therefore to be able to scale the reference ellipsoid according to the specified distance in the specified direction.

First, we must figure out which ellipsoid will be supporting the reflector in the specified direction $\mathbf{m}_0$, so that we know which ellipsoid we need to choose as reference ellipsoid. An ellipsoid is said to be *supporting* in a given direction if that ellipsoid is part of the final reflector surface in that direction. In other words, an ellipsoid is supporting when it is the closest one to the source along a given direction for a crossing reflector or the farthest one from the source for a non-crossing reflector. Since we do not know a priori the final shape of the reflector, we need to rely on some rules of thumb and observations.
Figure 3.10. Choice of a reference ellipsoid to set the reflector size. With a crossing reflector each ellipsoid facet is located on the opposite side from its corresponding target point.

The atlas of reflectors in Section 3.5 provides valuable insight into what generic shapes are expected for typical target distributions. When a uniform target is centered on axis, the supporting ellipsoid in the direction \((x,y,z) = (0,0,-1)\) is the ellipsoid corresponding to the target point at the center of the target, as shown in Figure 3.10a. So if one wants to set the source-apex distance, the reference ellipsoid should be the one corresponding to the central target point. The scaling parameters \(d_0\) or \(a_0\) (depending on the chosen parameterization) can then be calculated by solving Eq. 3.1 or Eq. 3.2 given the two foci and the polar radius \(\rho_0\) in direction \(\mathbf{m}_0\). We find

\[
\begin{align*}
\alpha \leq 0, \quad d_0 &= \frac{v_0 - \alpha \rho_0 + |\alpha| \sqrt{v_0^2 - 2\alpha v_0 \rho_0 + \rho_0^2}}{v_0/\rho_0 - 2\alpha}, \\
\alpha > 0, \quad d_0 &= \frac{v_0 - \alpha \rho_0 - |\alpha| \sqrt{v_0^2 - 2\alpha v_0 \rho_0 + \rho_0^2}}{v_0/\rho_0 - 2\alpha}, \\
a_0 &= \frac{1}{2} (\rho_0 + \sqrt{v_0^2 + \rho_0^2 - 2\rho_0 v_0 \alpha}),
\end{align*}
\] (3.4) (3.5)
where $v_0 = |v_0| = 2c = ST_0$ is the distance between the two foci of the reference ellipsoid (the source $S$ and the target point $T_0$), and $\alpha = m_0 \cdot \hat{v}_0$. If we wish to set the source-rim distance, we can leverage the system symmetry to figure out which ellipsoid will be supporting in the desired direction. For instance, for the uniform square target shown in Figure 3.10b, the supporting ellipsoid in the direction $(0,1,0)$ corresponds to the lower edge target point along the $y$-axis for a crossing reflector, and to the upper edge target point for a non-crossing reflector. In the general case (no particular symmetry), it can be useful to run the algorithm with a small number of rays and a small number or ellipsoids to visualize the overall reflector geometry and make an educated guess at which ellipsoid is the most likely to be supporting in the desired direction.

**Initialization of the focal parameters**

The starting point must fulfill the requirement that all flux is collected by the reference ellipsoid. Oliker provides lower and upper bounds for the focal parameters $d_i$ so that all ellipsoids are either smaller or larger than the reference ellipsoid [73]. We assume the reference ellipsoid to be $E_0$ and thus the focal parameter $d_0$ is fixed. For all $d_i (i > 0)$, we have

$$\alpha_{\text{min}} d_0 < d_i < \alpha_{\text{max}} d_0,$$

with $\alpha_{\text{min}} = \frac{1 - \gamma}{2}$ and $\alpha_{\text{max}} = \frac{2}{1 - \gamma_0 \sqrt{1 + \frac{d_0^2}{M^2} - \frac{d_0}{M}}}$,

where $\gamma_i = \max_{\Omega}(m \cdot \hat{v}_i)$ is the maximum of the dot product between the unit vectors $m$ (set of source ray directions over the reflector collection angle $\Omega$) and the unit vector $\hat{v}_i$ (axis direction
of ellipsoid $E_i$), as defined in Figure 3.2. $\gamma = \max_i \gamma_i$ is the maximum of all $\gamma_i$ values. $M = \max_i |v_i|$ is the maximum distance between the source $S$ and any target point $T_i$.

In order to generate a crossing reflector, all ellipsoids must initially be larger than the reference ellipsoid, so that according to the intersection rule all flux is collected by the reference ellipsoid. All focal parameters $d_i (i > 0)$ should therefore initially be set equal to the upper bound $\alpha_{max} d_0$. Conversely, in order to generate a non-crossing reflector all focal parameters $d_i (i > 0)$ should initially be set to the lower bound $\alpha_{min} d_0$.

This starting point is pessimistic since it takes into account the full ellipsoids, instead of only considering the “useful” portion of the ellipsoids within the reflector collection angle. However, a more accurate estimate does not necessarily bring significant benefit since all ellipsoids are quickly scaled according to their used region during the first cycles of the algorithm.

It is also worth mentioning that we can leverage system symmetries to reduce computation time. For instance, if both the source and the target share quadrant symmetry, only one fourth of the reflector needs to be computed.

**Iterative scaling of the focal parameters**

The initial scaling algorithm proposed by Kochengrin goes through the following steps. The reference ellipsoid, which is fixed, is assumed to be $E_0 (i = 0)$. For non-crossing reflector geometries negative increments must be used instead of positive increments.
Step 1  Set increments to $\Delta d_i = d_i / 3$.

Step 2  For all $i > 0$, set focal parameters to $d_i = d_i - \Delta d_i$.

Step 3  Evaluate target distribution (see Section 3.3).

Step 4  For all $i > 0$ satisfying $\Gamma_i, \text{measured} > \Gamma_i, \text{desired}$ halve increment $\Delta d_i = \Delta d_i / 2$ and set $d_i = d_i + \Delta d_i$. Then go to step 3. If for all $i > 0$, $\Gamma_i, \text{measured} < \Gamma_i, \text{desired}$ then start a new cycle from step 1.

Each cycle “removes” some flux from the reference ellipsoid and spreads it to the remaining ellipsoids. At all times the reference ellipsoid must remain above its target flux while all other ellipsoids must remain below their target flux. This constraint ensures convergence, but it also spreads error unevenly among ellipsoids. If the algorithm is interrupted “early” before convergence to a solution, then error tends to be larger in the reference ellipsoid than in the other ellipsoids. This phenomenon is illustrated in Figure 3.11, where we plot error in the reference ellipsoid and peak error among all other ellipsoids during an algorithm run. We define the error $e_i$ corresponding to ellipsoid $E_i$ as the absolute value of the difference between the desired target value and the measured target value at $T_i$,

$$e_i = \left| \Gamma_i, \text{measured} - \Gamma_i, \text{desired} \right|.$$

(3.7)

When the algorithm has finally converged, the only remaining error is caused by the accuracy of the target evaluation method and is now spread evenly among all ellipsoids, as shown in Figure 3.12. In fact, the remaining error is often significantly below the statistical noise limit.
Figure 3.11. Comparison of the peak error in the reference ellipsoid vs. all other ellipsoids during an algorithm run (log scale). Error is defined in Eq. 3.7 as the absolute value of the difference between measured target value and desired target value.

Figure 3.12. Comparison of error distribution among ellipsoids before and after an algorithm run. The reference ellipsoid originally has the largest error. After convergence, error is randomly distributed and is below the Monte-Carlo statistical noise limit.

Kochengrin proposed to use the Nelder-Mead algorithm to spread the error and accelerate the convergence rate [72]. However, this approach does not ensure convergence and depends on the choice of the initial reflectors. One drawback of the original algorithm is that it resets the same
initial increments at each new cycle. But as Figure 3.13 shows, the change in focal parameter tends to decrease as the number of cycles increases ($d_0$ corresponds to the reference ellipsoid and thus always remains constant).

![Figure 3.13. Evolution of the values of the focal parameters during an algorithm run.](image)

This means that the starting increments are in general too large and many iterations are necessary to scale them to their appropriate value. We devised a modified algorithm that provides better increment scaling.

**Step 1** Set initial increments to $\Delta d_i = (\alpha_{\text{max}} - \alpha_{\text{min}}) d_0 / 4$.

**Step 2** For all $i > 0$, set focal parameters to $d_i = d_{i-1} - \Delta d_i$.

**Step 3** Evaluate target distribution (see Section 3.3). If all target values are identical to the previous iteration, then double all increments $\Delta d_i = 2\Delta d_i$.

**Step 4** For all $i > 0$ satisfying $\Gamma_{i, \text{measured}} > \Gamma_{i, \text{desired}}$ halve increment $\Delta d_i = \Delta d_i / 2$ and set $d_i = d_{i-1} + \Delta d_i$. Then go to step 3. If for all $i > 0$, $\Gamma_{i, \text{measured}} < \Gamma_{i, \text{desired}}$ then go to step 5.
**Step 5**  For all \( i > 0 \), scale up increments to \( \Delta d_i = \kappa \Delta d_i \) (\( \kappa > 1 \)), then start a new cycle from step 2.

In this case each new cycle uses a scaled up version of the increments obtained at the end of the previous cycle (step 1 in which increments are initialized is only used once during the very first cycle). The factor \( \kappa \) controls by how much increments are scaled up from one cycle to the next. Figure 3.14 shows the effect of \( \kappa \) on the convergence speed. We found that a good value for \( \kappa \) is 1.25. If \( \kappa \) is too large then convergence tends to slow down. On the other hand, when \( \kappa \) is too small then the algorithm may stall before reaching the noise limit. For this reason, we check in step 3 that new increments have induced a change in target values. If this is not the case, then all increments are doubled until target values start changing. In Figure 3.14 we also show the performance of the original algorithm with no increment scaling. We have found that the required number of merit function evaluations is on average decreased by a factor of 4 with our modified algorithm, which results in a dramatic gain in speed.

Figure 3.14. Evolution of the normalized RMS error during an algorithm run for various increment scaling factors \( \kappa \), and for no increment scaling (original unmodified algorithm). Increment scaling dramatically improves convergence speed.
Termination criteria

Multiple criteria can be used to terminate the algorithm: maximum number of iterations, minimum improvement between two consecutive cycles, maximum RMS error or maximum peak error. The error is calculated for each target point as the absolute value of the difference between the measured and the desired target value, as defined in Eq. 3.7. We define the overall RMS error $\varepsilon_{\text{RMS}}$ of a reflector by

$$
\varepsilon_{\text{RMS}} = \sqrt{\frac{1}{N} \sum_{i=0}^{N} \varepsilon_i^2} = \sqrt{\frac{1}{N} \sum_{i=0}^{N} (\Gamma_i,\text{measured} - \Gamma_i,\text{desired})^2},
$$

where $N$ is the total number of ellipsoids, $\Gamma_i,\text{measured}$ is the measured value at target point $T_i$ and $\Gamma_i,\text{desired}$ is the desired value at target point $T_i$. This algorithm can in theory converge to an ideal solution where all target goals are met [71]. In practice, the accuracy of the resulting reflector is limited by the method used to evaluate the target. So if we let the algorithm run until it reaches stagnation (the algorithm ensures that RMS error can only decrease or stagnate), we can expect the reflector accuracy to match the accuracy of the target evaluation method, as detailed in Section 3.3.

Since the algorithm uses a discrete set of target point, it is often desirable to use as many points as possible in order to get a good approximation of the desired continuous target distribution. However, using a large number of target points can quickly become computationally prohibitive. Increasing the number of target points has two effects. First, it increases the required number of algorithm iterations to reach convergence, because it needs to scale more ellipsoids at the same time. Second, and most importantly, it increases the required time to evaluate the merit function. For instance, when using Monte-Carlo ray tracing to evaluate the target values, more rays need
to be traced to maintain accuracy and each ray must be tested for intersections with more ellipsoids. Because of these two combined effects, computation time increases quadratically with the number of target points, as shown in Figure 3.15.

Figure 3.15. Relative convergence time vs. number of target points.

3.2.2 Algorithm B: Least squares optimization

The major drawbacks of the previous algorithm are its slow rate of convergence and the error buildup in the reference ellipsoid. These concerns can be partially resolved with an alternative approach based on least square optimization. The main difference between the two approaches is the way focal parameters are scaled. In this case all focal parameters except the reference ellipsoid are given as input variables to an optimization algorithm that attempts to minimize the measured error at each target point. In order to do so, the optimization routine perturbs the values of the focal parameters and, based on the induced change in target values, predicts new focal parameters for all ellipsoids. Using optimization can be seen as pushing the increment scaling approach described in the previous section one step further: the optimization engine attempts at
each iteration to find the best increment value based on the estimated sensitivity of the target values on the focal parameters.

**Initialization of the focal parameters**

As with most local optimization routines, convergence towards a solution depends on how good the starting point is. The optimization routine can only evaluate the sensitivity of a target value to a change in focal parameter if the corresponding ellipsoid is supporting to the reflector. An ellipsoid is said to be *supporting* in a given direction if that ellipsoid is part of the final reflector surface in that direction. Whenever an ellipsoid is non-supporting, it is not part of the reflector shape and thus no light is reflected towards its corresponding target point. Consequently, as long as an ellipsoid remains non-supporting the corresponding target value will be zero and the sensitivity of the target value to the focal parameter cannot be evaluated. All ellipsoids must therefore be supporting to the reflector at the starting point, and must remain supporting to the reflector at all times. Target points should always have non-zero values to prevent optimization from getting stalled.

In order to get a starting point in which all ellipsoids are supporting to the reflector, one can use algorithm A and stop the iterative process once all target values are non-zero. In fact, only a small number of rays are required to obtain such a starting point. In theory, only one ray per ellipsoid would be required to ensure that ellipsoids are supporting. In practice, one may want to use more rays to get a starting point closer to the desired solution. In all cases, because the number of rays is very small, obtaining a starting point can be very fast.
Iterative scaling of the focal parameters

We tested this approach with the *LightTools* optimization engine, which uses least squares optimization. All focal parameters were set as variables, except the focal parameters corresponding to the reference ellipsoid, which sets the overall size of the reflector. The flux values measured at each target point were used as merit functions. In Figure 3.16, the evolution of the overall RMS error is plotted against the number of merit function evaluations (the number of times the target distribution is evaluated) for a small number of ellipsoids. Algorithm A is first used to obtain a starting point, which is the reason why the two curves overlap during the first 50 merit function evaluations. The two curves then split and show that the optimization algorithm provides a higher convergence speed.

Figure 3.16. Comparison of the convergence speed of the two proposed shape generation algorithms. In this example 9 target points were used.
During each cycle the algorithm successively perturbs each focal parameter by an amount $\Delta d_i$ and evaluates its impact on all target points $T_i$ in order to devise a new set of focal parameters. For $N$ target points, $N^2$ merit function evaluations are thus required, which can be computer intensive when the number of target points increases, and can thus make the optimization approach less advantageous. Specialized algorithms that make simplifying assumptions can be used to reduce the number of MF evaluations.

**Termination criteria**

It is important to stop iterating when the RMS error has reached the precision of the target evaluation method used (for instance, the statistical noise floor for Monte-Carlo estimation), as detailed in Section 3.3. Otherwise, we could end up optimizing noise and let the optimization engine run in vain.

<table>
<thead>
<tr>
<th>Table 3.1. Comparison of algorithm A and algorithm B.</th>
<th>Algorithm A (modified Kochengrin)</th>
<th>Algorithm B (least squares optimization)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Advantages</td>
<td>• No starting point required.</td>
<td>• Fast convergence speed.</td>
</tr>
<tr>
<td></td>
<td>• Convergence to global minimum is guaranteed.</td>
<td></td>
</tr>
<tr>
<td>Drawbacks</td>
<td>• Slow convergence speed.</td>
<td>• Requires starting point where all target values are non-zero.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• May not converge to global minimum.</td>
</tr>
</tbody>
</table>
3.3 Target distribution evaluation

The target distribution must be evaluated multiple times during each iteration of the algorithm. In order to generate a reflector for a uniform square target it takes about 500 target evaluations with 25 target points, and more than 4,000 iterations with 100 target points (when using algorithm A described in Section 3.2.1). For this reason, the speed of the algorithm, as well as the accuracy of the result, are largely driven by the efficiency of the target distribution evaluation method. Given a set of ellipsoids with different foci and focal parameters, we need to find how much flux is collected by each ellipsoid. Computing the exact intersections between ellipsoids is a difficult and costly operation. Instead, we propose two approaches: Monte-Carlo ray tracing and contour detection of the ellipsoid facets.

3.3.1 Monte-Carlo estimation method

Most illumination software packages use Monte-Carlo ray tracing to evaluate flux propagation in a system. The general idea is to trace a set of rays through the system and collect the rays on a receiver. The density of the rays on the receiver is then used to estimate intensity or illuminance, as detailed in Appendix B. In our case, we only need to determine the supporting ellipsoid for each ray. If a given ray is reflected by ellipsoid \( E_i \), we know without further calculation that the ray will be reflected towards its corresponding target point \( T_i \). If the reflector has crossing geometry, then the ellipsoid that reflects the ray is the one closest to the source. For each ray direction we thus compute the distance between the source and each ellipsoid using Eq. 3.1. The supporting ellipsoid then corresponds to either the minimum or maximum computed radius, depending on the reflector geometry (crossing or non-crossing):
Crossing geometry \[ \rho(m) = \min_i (\rho_i(m)), \quad i = 0, \ldots, N, \]

Non-Crossing geometry \[ \rho(m) = \max_i (\rho_i(m)), \quad i = 0, \ldots, N. \]

As with any Monte-Carlo ray tracing method, the generation of the source rays must be done according to the intensity distribution of the source. Two approaches are possible: sample rays uniformly in solid angle space and assign weights according to the intensity distribution, or set equal weights to all rays and generate more rays where intensity is higher. In our case, it can be more advantageous to use weighted rays so that the reflector surface is sampled uniformly. We use stratified sampling of the source rays to ensure good sampling uniformity while avoiding artifacts that may arise with sampling over a rectangular grid. More details on sampling techniques for Monte-Carlo ray tracing are given in Appendix B.2.

**Reflector accuracy and noise estimation**

The accuracy of Monte-Carlo methods depends on the number of rays traced. Tracing more rays takes more time, but it yields a better estimate of the target distribution and thus a more accurate reflector surface. The statistical noise inherent to Monte-Carlo ray tracing can be estimated by

\[
\varepsilon_{\text{noise}} = \frac{1}{\sqrt{N_{\text{rays}}}},
\]

where \(N_{\text{rays}}\) is the number of rays per ellipsoid, assuming an even distribution of the rays among ellipsoids. More details on Monte-Carlo simulations are given in Appendix B. Once target values are below the estimated peak statistical error, the iterative process should be stopped, since the measured values are then dominated by statistical noise.
Improving evaluation speed

Several minor enhancements can help improving the speed of target evaluation. First, the eccentricity in Eq. 3.1 is computed only once after the focal parameter has been updated, which avoids unnecessary computations every single time a ray is traced. Second, the task of evaluating the target distribution can easily be split into independent processes, each of which handles a different subset of rays. Evaluation can thus be multi-threaded and take advantage of machines with multiple cores. A comparison of the performance obtained with different number of threads is given in Figure 3.17. With 4 threads, the computation time for an algorithm run is about 3 times faster than with a single thread.

![Computation time graph](image)

Figure 3.17. Effect of multi-threading on computation time.

We give next some suggestions for future improvement of the algorithm performance. These ideas were not implemented and remain to be further explored. Instead of keeping the same set of rays throughout the algorithm run, the number of rays can be gradually increased as the algorithm gets closer to the desired solution. By starting with a small number of rays and only refining the accuracy when needed, significant computation time can be saved. Another possible
way to save time is to track which rays are “active”: as the algorithm progresses it reaches a state in which changes from one cycle to the next only have minor effects on the ellipsoids. Consequently, most rays will keep hitting the same ellipsoid, and only the rays close to the edges of each ellipsoid facet are relevant to the measured change in flux at each target point. This means, at least in theory, that only a small subset of rays would need to be traced at each cycle. Besides, we can leverage the knowledge of the sequence of intersected ellipsoids to predict which ellipsoids are in the neighborhood of each ray. Instead of computing the intersection of each ray with each ellipsoid in the system, we only consider ellipsoids in the neighborhood of each ray, since they are the ones that are likely to intersect the ray after a small change in focal parameter. In other words, not only can we trace specific rays, but we can also test only specific intersections with relevant ellipsoids in the neighborhood of each ray, thus saving significant computation time.

3.3.2 Contour detection method

An alternative method to compute the flux collected by each ellipsoid consists in computing the solid angle subtended by each facet. If the source is isotropic, then the collected flux simply corresponds to the source intensity $I$ multiplied by the subtended solid angle $\omega$. The key difficulty is to compute the value of an arbitrarily shaped solid angle. Fortunately, the solid angle can easily be computed from its projected contour on a reference sphere using Eq. A.16, as detailed in Appendix A. So in the end, we just need to retrieve the contour of each ellipsoid facet of the reflector.
We give a brief outline of one possible approach to retrieve facet contours. From an interior facet point, we move in an arbitrary direction until we find the edge of the facet. From there, we use an algorithm that “walks” along the facet contour until it loops back to its original starting point, as illustrated in Figure 3.18. Each contour point is found using a bisection method that oscillates between the inside and the outside of the facet, while reducing the gap at each step. The contour detection approach requires as a first step to obtain at least one ray hitting each facet in the reflector. In order to do so, rays generated in random directions can be traced until at least one ray reaches each facet.

![Contour detection method](image-url)

Figure 3.18. Contour detection method. From an interior point, we move until we find the facet edge and then proceed along the facet edge using an iterative bisection search at each step.

The contour detection method has significant overhead compared with the Monte-Carlo ray tracing approach. Indeed, a starting ray must be found for each facet, and the algorithm must iteratively find the facet edge up the specified level of accuracy. When the number of contour points is small, then the corners of the facet are likely to be chopped and the corresponding measured solid angle will then be underestimated. So for small number of points the contour
detection approach does not present significant benefits in terms of speed and accuracy compared to the Monte-Carlo ray tracing approach. On the other hand, with large number of contour points the facet contour can be well approximated and the measured solid angle will be estimated with higher accuracy than with the Monte-Carlo approach.

In the end, the contour detection approach appears to only be valuable when high accuracy is required and a large number of rays need to be traced. Most of the cases we encountered did not require the level of accuracy that would justify the use of the contour detection method, so we focused our research efforts on the Monte-Carlo approach, which is also easier to implement robustly.

3.4 Surface representation and smoothing

3.4.1 Surface representation

Representing and visualizing the generated freeform reflector surface is not as straightforward as it may first seem. Generated reflectors are made of a finite number of ellipsoids. If the reflector is crossing, then the reflector surface corresponds to the inner envelope of the ellipsoid collection. If the reflector is non-crossing, then the reflector surface is the outer envelope of the ellipsoid collection. In CAD software using constructive solid geometry, the reflector shape can be obtained using Boolean operations. A crossing reflector corresponds to the intersection of all the ellipsoids, while a non-crossing reflector corresponds to the union of all the ellipsoids. The resulting solid then needs to be trimmed according to the collection angle of the reflector. Figure 3.19 shows the original full ellipsoids and the resulting shape after intersecting and trimming.
The final reflector surface is made of patches of ellipsoids, which we shall refer to as facets. Overall, the reflector surface is continuous but slope discontinuities occur at the edge of each facet (the surface has $C_0$ continuity).

![Figure 3.19. Generation of the reflector surface from a set of 9 ellipsoids](image)

In practice this method does not work well even for modest number of ellipsoids ($N > 25$). The algorithms used to calculate intersections between ellipsoids in CAD software can become time consuming, and the booleaning operation may encounter difficulties when the number of ellipsoids becomes too large. The ellipsoid slopes are often very similar where they intersect, thus creating instabilities in the algorithms. One possible workaround consists in retrieving the approximated contour of each facet with the contour detection techniques detailed in Section 3.3.2. Each ellipsoid is then trimmed independently according to the retrieved contour, and all facets are finally stitched together.

Since it proves difficult to directly use an analytical surface representation, we can instead sample the surface into a set of discrete points. From there, we can convert the set of points into
a polygonal mesh or interpolate the points with functions such as NURBS. Let’s have a closer look at these two alternatives. Examples of polygon mesh and NURBS interpolation are given in Figure 3.20. A polygon mesh is made of flat facets; therefore we need a lot of them to obtain a good fit to the original surface. Scattered data points can easily be converted into a polygon mesh using triangulation techniques. On the other hand, NURBS interpolation produces a smooth surface which can easily be deformed by moving control points and knot vectors. However, it is difficult to convert a set of scattered data points into a NURBS (handling the surface boundary becomes especially difficult). NURBS interpolation requires a rectangular grid of points in a given \((u,v)\) surface parameterization. Since our surface description is based on spherical coordinates, it is convenient to sample the reflector along a \((\theta, \phi)\) grid. However our sampling resolution will be higher closer to the pole. In the end, we opted for the use of NURBS, because of the variety of existing tools and the fact that they can be more easily exported and imported between various software packages.

![Polygon mesh and NURBS](image)

Figure 3.20. Exported reflectors made of (a) a triangular mesh and (b) a NURBS surface.
All these methods are subject to a certain degree of uncertainty arising from the interpolation process. The type of interpolation or triangulation used has an impact on the generated surface and hence on the target distribution. As a general rule of thumb, the higher the number of sampled points is, the closer the generated surface is to the initial surface. In some cases the smoothing effect produced by interpolation can be leveraged. Indeed, our goal is not necessarily to generate a surface as close as possible to the initially generated surface, but instead to obtain a target distribution as close as possible to the desired one. Besides, there is no such thing as slope discontinuities in the physical world. When manufactured, edges between facets would be filleted and thus spread some of the incoming light. Figure 3.21 shows the effect of sampling on the measured target distribution. In this case and everywhere else in this dissertation, the initial generated surface is sampled along a \((\theta, \phi)\) grid, converted into a NURBS surface, exported to an IGES file, and finally imported into the illumination design software \textit{LightTools}\textsuperscript{®} for analysis.

![Figure 3.21. Target distributions obtained with various surface sampling accuracies for a uniform 10x10 target grid. The sampling grid accuracy in spherical coordinates is indicated as number of points along \(\theta\) \times number of points along \(\phi\).](image)

Ideal ellipsoid facets would produce perfect points on the target. However, as the sampling grid accuracy decreases the surface become less accurate and some target points get blurred or even
degenerate into lines, as in the case of the 15×75 grid. Because we use spherical coordinates, sampling is denser in the regions of the reflector surface close to the poles. This explains why target points are more accurately rendered at the center of the target than close to the edges.

3.4.2 Surface interpolation

The main drawback of the Oliker algorithm is that it generates a reflector shape for a discrete target instead of a continuous distribution. We must thus find a way to obtain the desired continuous target distribution from the discrete solution. This is typically done by interpolating or approximating the discrete data. In our case, we can proceed in two different manners: interpolating the polar radius of the surface or interpolating the focal parameters.

Our starting reflector is a surface made of ellipsoid facets. One can interpolate the surface by choosing a set of discrete points across the surface and then interpolate or approximate the discrete set of polar radii into a smooth continuous surface. The role of interpolation here is to spread the flux originally concentrated at each target point according to the desired target distribution. The performance of the interpolation process thus depends on how close we can get to the desired distribution. Standard interpolation methods however do not take into consideration the impact of the fitting on the target distribution. As a result, direct interpolation of the reflector surface often results in artifacts, especially close to the edges of the target. Figure 3.22 shows examples of such artifacts. In this case we sampled the original surface by taking one point at the center of each ellipsoid facet. The corresponding set of scattered points is then approximated in spherical coordinates by the Matlab routine gridfit [74]. Even a large number of
ellipsoids does not necessarily make artifacts fully disappear, as the example with 40x40 target points on the right of Figure 3.22 shows.

![Figure 3.22. Target distributions produced by interpolation of the polar radius of the reflector surface, for a uniform square target.](image)

An alternative strategy consists in interpolating the focal parameters instead of the polar radius. The solution produced by the Oliker algorithm is a discrete set of focal parameters $d_i$ (one for each target point $T_i$). One can interpolate this discrete set into a continuous focal parameter function $d(x,y)$ over the target domain. The reflector surface then corresponds to the inner or outer envelope of the corresponding family of ellipsoids, depending whether the reflector is crossing or non-crossing. A method to find the envelope of a family of conics was proposed in [75, 76]. Alternatively, one can use this method to increase the density of the target grid and then use interpolation of the reflector polar radius from a finite (but large) number of ellipsoids.

In all cases, standard interpolation techniques do not provide direct control of how flux is spread over the target, and thus often produce arbitrary results that differ from the intended target distribution, unless the number of ellipsoids is very high, which then requires prohibitively long
computation time. Chapter 4 proposes an alternative approach based on source-target maps to
generate smooth reflectors that produce continuous distributions in agreement with the desired
target distributions.

Interpolation could also be used to speed up the Oliker algorithm, by gradually increasing the
resolution of the target grid. For instance, a reflector with a low resolution target grid would first
be generated, would then be interpolated to twice the initial target grid resolution, re-optimized,
and so on, until the required performance is reached. Each newly interpolated reflector provides
a starting point that “jump-starts” the algorithm and thus reduces the computation time required
to reach a solution. This promising approach remains to be explored.

3.5 Atlas of generated reflectors
In this section we look at the effect of various system parameters on the reflector shape. The goal
is to give an idea of reflector shapes for typical system configurations and also to give some
insight on how the target distribution can be modified according to aesthetic constraints. The first
set of figures (from 3.22 to 3.26) uses a typical streetlight configuration: the source is placed
30 ft above the street and the target must be illuminated uniformly. The method of supporting
ellipsoid can be used for any source intensity distribution and any arbitrary target shape (not
necessarily flat, and not necessarily uniform), as Figure 3.26 and Figure 3.30 show. Two
reflector shapes can always be generated for the same target distribution, crossing or non-
crossing. Crossing geometry typically yields more compact shapes.
Figure 3.23. Effect of the reflector geometry (crossing / non-crossing) on the reflector shape.

Figure 3.24. Effect of the target size on the reflector shape.
Figure 3.25. Effect of the target aspect ratio on the reflector shape.

Figure 3.26. Example of a tilted target with non-constant \( z \) coordinates.
Figure 3.27. Effect of a shift of the target along the $x$-axis (off-axis target).

Figure 3.28. Effect of the collection angle on the reflector shape (crossing geometry).
A couple of rules of thumb can be useful to keep in mind. For crossing reflectors, a large target extent yields compact and deep reflectors, as shown in Figure 3.24. On the other hand, for non-crossing reflectors a large target extent yields wide and shallow reflectors. When the target is asymmetric, then a crossing reflector will tend to be narrow along the direction of the largest target extent and narrow along the direction of the smallest target extent (shape of a football...
ball). The exact opposite happens with non-crossing reflectors, as shown in Figure 3.25 (shape of a bowtie). When the target is shifted off-axis, crossing reflectors tend to become narrower in the direction of the shift, while non-crossing reflectors tend to become wider. Finally, the size of both crossing and non-crossing reflectors increases dramatically when the collection angle increases. Crossing reflectors become deeper while non-crossing reflectors become wider, as shown in Figure 3.28. For large collection angles, non-crossing reflectors can get so wide that they become impractical to use. On the other hand, crossing reflectors may self-block some of the reflected rays and thus not produce the expected target distribution as shown in Figure 3.31.

![Figure 3.31. Self-blockage of the reflected rays by the reflector. The rays in red are reflected a second time by the reflector and thus do not reach their expected target location. This situation often arises with large targets and large collection angles.](image)

### 3.6 Alternative shape generation method using linear programming

So far we have considered targets at finite distance from the source (near-field). In this case the target is defined as a prescribed illuminance distribution. When the target is defined as an intensity distribution (far-field), then the reflector shape can be formulated as a variational problem and can be solved using linear programming. This alternative approach was proposed independently by Wang [55] and by Oliker [77, 78] in 2003. As with the rotationally symmetric
case, far-field targets are somewhat simpler than near-field targets to deal with because the direction of the reflected rays does not depend on position. The philosophy of the variational method is as follow. Each source ray is considered as a “supply” and each target value as a “demand”. The algorithm tries to optimize the system so that each target point fulfills its “demand” by assigning the source rays to the various target points. In practice, the variational formulation of the problem consists in maximizing a function under a set of linear constraints, which can be stated in discrete form as

\[
\max \left( \sum_{i=0}^{M} I_S(s_i) u(s_i) + \sum_{j=0}^{N} I_T(t_j) v(t_j) \right),
\]

\[
u_i + v_j \leq -\log (1 - s_i \cdot t_j), \quad i = 0, \ldots, M, \quad j = 0, \ldots, N,
\]

where \(s_i\) and \(t_j\) are unit vectors that define a set of source ray directions and target ray directions, and hence implicitly define the reflector collection angle and the target distribution in a discrete fashion. In other words, \(s_i\) and \(t_j\) sample the source and the target regions. \(I_S(s_i)\) and \(I_T(t_j)\) correspond respectively to the intensity of the source in direction \(s_i\) and to the prescribed intensity of the target in direction \(t_j\). The variables \(u_i\) and \(v_j\) correspond to the natural logarithm of the polar radius \(\rho\) of the reflector along directions \(s_i\) and to the focal parameter \(d\) of the supporting paraboloids of the reflector with axes \(t_j\) (referred to as the focal function), thus

\[
\rho(s_i) = e^{u_i}, \quad i = 0, \ldots, M,
\]

\[
d(t_j) = e^{v_j}, \quad j = 0, \ldots, N.
\]

This set of equations is equivalent to a linear programming problem, which is often written in the standard form [79] as
\[
\begin{align*}
\max (c^T x), \\
A x \leq b.
\end{align*}
\]  

(3.12)

Following the standard form notations, \(c\) and \(x\) are both vectors of size \(M+N\) given by

\[
\begin{align*}
x &= \begin{bmatrix} u(s_0), \ldots, u(s_M), v(t_0), \ldots, v(t_N) \end{bmatrix}, \\
e &= \begin{bmatrix} I_S(s_0), \ldots, I_S(s_M), I_T(t_0), \ldots, I_T(t_N) \end{bmatrix},
\end{align*}
\]  

(3.13)

and \(b\) is a row vector of size \(MN\) given by

\[
b = \begin{bmatrix} b_{0,0}, b_{0,1}, \ldots, b_{0,N}, b_{1,0}, \ldots, b_{1,N}, \ldots, b_{M,N} \end{bmatrix}, \quad b_{i,j} = -\ln(1-s_i \cdot t_j). 
\]  

(3.14)

Finally, \(A\) is a matrix of size \((M+N) \times MN\), and is filled according to Eq. 3.10. The resulting matrix is sparse and only contains 0 and 1, which can be leveraged to improve the algorithm speed and reduce memory requirements. Importantly, we need to ensure that the total flux emitted by the source corresponds to the total flux on the target, which means that we must have

\[
\sum_{i=0}^{M} I_S(s_i) = \sum_{j=0}^{N} I_T(t_j). 
\]  

(3.15)

In practice we can simply normalize the target flux by the total source flux. Eq. 3.16 yields crossing reflectors. Non-crossing reflectors can be obtained in the same fashion by minimizing the same function, under the opposite set of linear constraints, by using

\[
\begin{align*}
\min \left( \sum_{i=0}^{M} I_S(s_i) u(s_i) + \sum_{j=0}^{N} I_T(t_j) v(t_j) \right),
\end{align*}
\]  

(3.16)

\[
u_i + v_j \geq -\log(1-s_i \cdot t_j), \quad i = 0, \ldots, M, \quad j = 0, \ldots, N.
\]

We can transform Eq. 3.16 into a maximization problem by replacing \(u_i\) by \(-u_i\) and \(v_i\) by \(-v_i\). In order to match the standard form, the direction of the inequality can be reversed by using \(-b\) instead of \(b\) and \(-A\) instead of \(A\). The shape of near-field reflectors depends on their scale.
because a different scale requires different surface normals to aim rays towards the correct target location. Far-field reflectors are free from this constraint, and thus the reflector scale has no importance: any scaled versions of a far-field reflector produce the same distribution in the far-field. However, we need to impose a constraint that sets the scale in our problem. Otherwise, there would be an infinite number of solutions. Wang suggests to set $u_0 = 0$, which is equivalent to setting $\rho(s_0) = 1$. Let’s now consider a practical example using a small number of points in order to understand how to set up the problem correctly. We will use only 4 source ray directions $s_0, s_1, s_2, s_3$ and 2 target directions $t_0$ and $t_1$, defined in Cartesian coordinates as

$$
\begin{align*}
s_0 &= \frac{1}{\sqrt{2}} (1, 0, -1), \\
s_1 &= \frac{1}{\sqrt{2}} (0, 1, -1), \\
s_2 &= \frac{1}{\sqrt{2}} (-1, 0, -1), \\
s_3 &= \frac{1}{\sqrt{2}} (0, -1, -1),
\end{align*}
$$

We thus have

$$
\mathbf{b} = [0, -\ln(2), -\ln(1.5), -\ln(1.5), -\ln(2), 0, -\ln(1.5), -\ln(1.5)].
$$

Additionally, the source is assumed to be isotropic and the target uniform. We must be careful to normalize the components of vector $\mathbf{c}$ according to Eq. 3.15. We thus have

$$
\mathbf{x} = [u_0, u_1, u_2, u_3, v_0, v_1],
$$

$$
\mathbf{c} = [1, 1, 1, 1, 2, 2].
$$
Solving this linear programming program with Matlab returns, for the crossing geometry,

\[ x = [0,-0.2877,0,-0.2877,0.6931,0.6931], \]

which fulfills the symmetry we would expect from the system. The corresponding reflector polar radii are then given by

\[ \rho(s_0) = e^0 = 1, \quad \rho(s_1) = e^{-0.2877} \approx 0.75, \quad \rho(s_2) = 1, \quad \rho(s_3) \approx 0.75. \]

As with the method of supporting ellipsoids, there is no limitation imposed on the source intensity distribution, the target intensity distribution, or the reflector collection angle. However, matrix \( A \) can quickly become large as the number of target points and source directions increase. For instance, for 1000 target points and 1000 source directions, the size of \( A \) becomes \( 2,000 \times 100,000 \), which is quite large. Nonetheless, numerous algorithms have been devised to handle large-scale problems and matrix \( A \) does not need to be stored in memory since it only contains 0 and 1 at pre-defined positions.

We tested the variational approach by implementing it in Matlab using the routine \textit{linprog} (linear programming function included in the optimization toolbox). Unfortunately, this generic function does not take into account the specific nature of our problem and ends up being extremely slow.
and memory intensive. Figure 3.32 shows an example of computed reflector points for a 15×60 source ray grid and a 3×3 target grid. It took about 80 s to compute, while the same configuration would take 0.2 s to compute with our implementation of the method of supporting ellipsoids. Using more points or rays quickly exceeded the memory limit of our system, so it is difficult to compare fairly this early implementation of the variational approach with the method of supporting ellipsoids. However, it would be insightful to further investigate a specific implementation that would leverage the scarcity of matrix $A$ and handle larger number of rays and target points.

![Figure 3.32. Example of a set of reflector points generated with the variational approach. In this example a 3×3 target point grid was used in conjunction with a 15×60 ray grid. The source is isotropic and the reflector has a crossing geometry.](image)
CHAPTER FOUR - SOURCE-TARGET MAPPING

In this chapter we investigate the relationship between the reflector shape and the way the source flux is mapped onto the target. Section 4.1 extends the mapping concepts used for systems with rotational or translational symmetry to systems without symmetry. However, these conventional mappings often require step discontinuities in the reflector shape to produce the correct target distribution. Section 4.2 establishes the integrability condition that must be fulfilled so that mappings can be achieved by smooth reflector surfaces. Section 4.3 proposes a method to devise integrable maps based on the Oliker algorithm. Section 4.4 shows how smooth reflectors can then be constructed from quasi-integrable maps. Finally, Section 4.5 examines relationships between integrable maps, reflector shapes and system parameters.

4.1 Mappings and shape generation

4.1.1 Mappings for systems with rotational symmetry

We saw in Chapter 2 that the design of a reflector for a system with rotational or translational symmetry typically undergoes two steps. First, the flux emitted by the source is mapped to the desired target distribution. Then, the reflector surface is generated in order to achieve the derived mapping. The mapping of the source rays to the target is often done by comparing the cumulative distributions of the source and the target. If the system has rotational symmetry, we can divide the intensity distribution of the source into annular solid angles that each contains the same amount of flux, as shown in Figure 4.1. Similarly, we can divide the target illuminance
distribution into rings that all contain the same amount of flux. A mapping can then be derived by matching each slice in the equi-flux source grid to a corresponding slice in the equi-flux target grid. This intuitive process is equivalent to solving the following equation

\[
\int_{\theta_0}^{\theta} I(\theta') \sin \theta' d\theta = \int_{y_0}^{y} E(y') y' dy'
\]

(4.1)

where \( I \) is the intensity of the source and \( E \) the desired illuminance distribution on the target. The left side of Eq. 4.1 corresponds to the collected source flux within an annular solid angle, and the right side corresponds to the target flux contained within a ring. From Eq. 4.1, we can obtain a relationship between the emission angle of the source rays \( \theta \) and the target location \( y \). In other words, we can express \( y \) as a function of \( \theta \) so that

\[
y = f(\theta)
\]

(4.2)

where \( f \) is a continuous function defined over the collection angle of the reflector.

![Figure 4.1. Source-target mapping for a system with rotational symmetry.](image-url)
Now that a mapping has been derived, we need to generate a reflector curve that achieves that mapping. For systems with rotational or translational symmetry only a single reflector curve needs to be generated; the full reflector can then be obtained by sweeping the curve around the axis of symmetry. Figure 4.2 illustrates how a simple geometrical construction can be used to generate the reflector curve. From a starting point \( P_0 \) (the apex of the reflector for instance), we can compute the required surface normal \( \mathbf{N}_0 \) so that the ray emitted at \( \theta_0 \) reached its corresponding target point at \( y_0 \). From there, we create a plane perpendicular to \( \mathbf{N}_0 \) at \( P_0 \) and find its intersection \( P_1 \) with a ray emitted at the next angle \( \theta_1 \). We can then compute the required normal \( \mathbf{N}_1 \) at this point, and repeat this process, until the entire curve is completed.

![Geometrical construction of a reflector curve in order to achieve a specified source-target mapping.](image)

The method shown in Figure 4.2 is effective but may generate some small slope error when the generated points \( P_i \) are interpolated into a continuous curve, because the interpolated curve will not exactly have the expected surface normal \( \mathbf{N}_i \) at \( P_i \). This minor slope mismatch can typically
be neglected when the number of generated points is large, since it does not accumulate over the generation process. Elmer proposed a modified generation method that avoids surface normal discrepancies in the interpolated curve [15]. In Figure 4.4 we detail a version of Elmer’s method that was extended to handle non-coplanar reflected rays. All the reflector shapes presented in this chapter were generated using this construction technique.

4.1.2 **Mappings for systems without symmetry**

When a system has neither rotational nor translational symmetry, the problem cannot be reduced to a 2D problem in a plane. Systems with no symmetry involve skew rays, and the flux mapping must now be formulated as a 3D problem. Instead of Eq. 4.1, for a system with no symmetry we now have

\[
\int_{\theta_0}^{\theta} \int_{\phi_0}^{\phi} I(\theta', \phi') \sin \theta' d\theta' d\phi' = \int_{x_0}^{x} \int_{y_0}^{y} E(x', y') dx' dy'
\]

where \( I \) is the intensity of the source and \( E \) the desired illuminance distribution on the target. The spherical coordinate system conventions used for the source throughout this chapter are defined in Figure 4.3.

![Spherical coordinate system angle conventions.](image)

Figure 4.3. Spherical coordinate system angle conventions.
Figure 4.4. Elmer’s method [15] (extended to 3D) for the generation of reflector curves.
Eq. 4.3 is a classical topology problem of mapping a 2D manifold (source intensity distribution) onto another 2D manifold (target illuminance distribution). Ideally, we would like to obtain relationships between the source coordinates and the target coordinates, such as

\[
x = f(\theta, \phi) \\
y = g(\theta, \phi)
\]  

(4.4)

where \(f\) and \(g\) are both continuous functions over the collection angle of the reflector. There is no general method to solve this problem. However, there are specific cases where the system symmetry allows separating variables and finding a solution to Eq. 4.4.

Figure 4.5. Source-target mapping derived using symmetry considerations.

Let’s consider the example depicted in Figure 4.5. In this example the reflector collects light over a hemisphere and the desired target is a uniform square. We assume that the point source is isotropic. Similarly to the rotationally symmetric case, we start by slicing both source and target distributions into equi-flux slices (blue contours). However, since now the system is not invariant
along $\phi$, we also need to slice the source and target distributions radially (green contours). We end up with a source grid and a target grid in which each bin contains the same amount of flux. If the source is isotropic then each bin in the source grid must have the same solid angle. Analogously, if the target is uniform then each bin of the target grid must have the same area. The example shown in Figure 4.5 is such a case where the source is isotropic and the target is uniform, and hence all source and target bins, despite having different shapes, have the same solid angle and the same area.

The mapping process based on equi-flux grids we just described can be put into an analytical form for a generic uniform rectangular target and an isotropic or Lambertian source emission pattern. We assume that the reflector collection angle is a cone delimited by $\theta_{\text{min}}$ and $\theta_{\text{max}}$. The mapping relationships between the source grid coordinates $(\theta, \phi)$ and the target grid coordinates $(x, y)$ are given by

$$
0 \leq \phi < \frac{\pi}{4} \quad x = x_{\text{max}} f(\theta) \quad y = y_{\text{max}} f(\theta) g(\phi) \\
\frac{\pi}{4} \leq \phi < \frac{3\pi}{4} \quad x = x_{\text{max}} f(\theta) (2 - g(\phi)) \quad y = y_{\text{max}} f(\theta) \\
\frac{3\pi}{4} \leq \phi < \frac{5\pi}{4} \quad x = -x_{\text{max}} f(\theta) \quad y = y_{\text{max}} f(\theta) (4 - g(\phi)) \quad (4.5) \\
\frac{5\pi}{4} \leq \phi < \frac{7\pi}{4} \quad x = x_{\text{max}} f(\theta) (g(\phi) - 6) \quad y = -y_{\text{max}} f(\theta) \\
\frac{7\pi}{4} \leq \phi < 2\pi \quad x = x_{\text{max}} f(\theta) \quad y = y_{\text{max}} f(\theta) (g(\phi) - 8)
$$

with

$$f(\theta) = \sqrt{\frac{\cos^n \theta_{\text{min}} - \cos^n \theta}{\cos^n \theta_{\text{max}} - \cos^n \theta}} \quad \text{and} \quad g(\phi) = \frac{\phi}{\pi/4},$$

117
where \( n = 1 \) for an isotropic source and \( n = 2 \) for a Lambertian source. \( x_{\text{max}} \) and \( y_{\text{max}} \) are respectively the half-width and half-height of the target. We assume that \( \phi \) belongs to the [0,2\( \pi \)] interval. We used this set of equations to generate the target grid depicted in Figure 4.5. See Appendix D for the derivation of these mapping equations.

It is important to keep in mind that the mapping achieved by Eq. 4.5 is not unique. In the example shown in Figure 4.5, we could decide to rotate the target map by 90°, so that the rays emitted at \( \phi = 0° \) are now reflected towards the \( y \)-axis on the target (\( \phi = 90° \)) instead of the \( x \)-axis. In fact, the target map could be rotated by any arbitrary angle and still map to the same illuminance distribution. The number of possible mappings is therefore infinite in this case. While we did not prove rigorously that the number of possible source-target mappings is always infinite, it substantiates the claim that there are in general many source-target mappings that can achieve the same target distribution.

4.1.3 Shape generation and slope mismatch

Once a mapping between the source and the target has been derived, we need to generate a reflector shape that maps each source bin onto its corresponding target bin, as depicted by the rays in red in Figure 4.5. Using an approach similar to the rotationally symmetric case, we can generate a set of reflector curves at various \( \phi \) angles (green contours). These profiles create a skeleton of the reflector, and a continuous reflector shape can be obtained by interpolating the collection of profiles with a NURBS surface, as shown in Figure 4.6.
Figure 4.6. Generation of the reflector shape based on a source-target mapping for a uniform square target. Profile rib curves are generated at various angles $\phi$ (green contours). The rib curves are then interpolated into a smooth reflector using a NURBS surface. The target illuminance produced by the generated reflector departs significantly from the expected uniform square the reflector was originally designed for.

However, when this method is used to generate a reflector for the uniform square target mapping shown in Figure 4.5, we obtain the target illuminance distribution shown on the right in Figure 4.6. That distribution is far from the expected uniform square, which means that the mapping achieved by the reflector is not the one we thought we designed it for. So what went wrong? Successive points along each green profile were calculated so that the resulting surface normals are correct in the direction of the integration path (along the green contours), but nothing guarantees that the surface normals will be correctly oriented in the direction orthogonal to the integration path (along the blue contours). As shown in Figure 4.7, the component of the surface normals along the blue $\phi$ contours is determined by the relative position of adjacent profiles, which appears to be erroneous in this case.
Figure 4.7. Slope mismatch arising during reflector generation. When the reflector surface is interpolated from a set of profiles along $\theta$, the component of the surface normals along $\phi$ may not match initial assumptions.

4.1.4 \textit{Wedge reflectors that minimize slope mismatch}

Based on this observation, we decided to use an alternative construction technique. We first generate a single reflector profile along $\theta$ (green curve in Figure 4.8). From each point on that initial profile we then generate a small reflector stripe along the $\phi$ direction (blue curves in Figure 4.8). Our goal here is to control the slope in the $\theta$ direction by first integrating the profile along $\theta$, and then to control the slope in the $\phi$ direction by generating reflector stripes along $\phi$. After interpolating the generated points into a smooth surface with a NURBS, we obtain a reflector wedge, as shown in Figure 4.8 on the right. This wedge generation process is repeated multiple times so that multiple wedges are created over the full extent of the reflector, as shown in Figure 4.9.
Figure 4.8. Wedge generation method that minimizes slope errors by integrating successively along $\theta$ and $\phi$. The green curve corresponds to integration from the reflector apex along $\theta$ and the blue curves correspond to integration along $\phi$. The set of generated points is then interpolated with a NURBS surface into a smooth wedge, as shown on the right.

This construction process does not guarantee that the final reflector shape will be continuous. In the case of a uniform square target shown in Figure 4.9, there are step discontinuities from one wedge to the next. However, this approach ensures that surface normals are correctly oriented, and we indeed obtain a target distribution much closer to the desired uniform square. Step discontinuities produce noticeable artifacts in the target distribution, which tend to become less noticeable as the number of wedges increases. Similar approaches have been investigated in [51, 80], where reflectors are divided in multiple continuous patches (not necessarily wedges). The size and shape of the patches is usually chosen in order to balance the artifacts created by the step discontinuities and the need to minimize slope errors in the generated surface.
Figure 4.9. Wedged reflectors and their corresponding target illuminance distributions. Wedged reflectors avoid slope mismatch arising when trying to generate a smooth reflector shape from the mapping shown in Figure 4.5. Generating more wedges minimizes artifacts in the target distribution but can make it difficult to manufacture the shape accurately.
4.2 The integrability condition

While the wedged reflector provides a solution close to the desired result, the remaining artifacts in the distribution can be undesirable and a shape with many step discontinuities can be difficult to manufacture accurately. As we will see, there is an infinite number of different ways to map the source flux onto the target and obtain the same target distribution. However, only a small subset of these mappings can be achieved with a smooth continuous surface. The mapping we used in the previous section, while intuitive, happens to be one of the many mappings that cannot be achieved by a continuous surface.

In the previous section we integrated the reflector shape in the \( \theta \) direction (along the green contours) and in the \( \phi \) direction (along the blue contours). In fact, we could choose any arbitrary integration path from any given starting point to generate the reflector shape. In order for the reflector shape to be continuous, integrating the surface normals along any closed integration path should yield a closed curve. In other words, if we integrate the surface normals over a closed loop we should end up where we originally started. If there is a difference in height between the starting and ending points, then no continuous surface can fit the given set of surface normals. Mathematically, this continuity constraint can be stated with a contour integral as

\[
\oint_C \mathbf{N} \cdot d\mathbf{l} = 0, \tag{4.6}
\]

where \( C \) is an arbitrary closed contour over the surface, \( \mathbf{N} \) is the field of surface normals and \( d\mathbf{l} \) is an infinitesimal displacement along the source grid. Using Stokes’ theorem, Eq. 4.6 is can be rewritten as

\[
\iint_S (\nabla \times \mathbf{N}) \cdot d\mathbf{S} = \iint_S (\nabla \times \mathbf{N}) \cdot \mathbf{N} \, d\mathbf{s} = 0, \tag{4.7}
\]

123
which is true if $\mathbf{N}$ is defined over a simply connected region [81]. Since this constraint applies for any integration path on the surface, the condition for surface continuity is equivalent to

$$\mathbf{N} \cdot (\nabla \times \mathbf{N}) = 0. \quad (4.8)$$

This constraint is often referred as the integrability condition or surface smoothness constraint in computer graphics [82]. It is often used in surface reconstruction algorithms such as shape from shading to ensure consistency in the data. In spherical coordinates, the reflector surface is defined as a 2-parameter function

$$r = f(\theta, \phi), \quad (4.9)$$

where $r$, $\theta$ and $\phi$ are the standard spherical coordinates, and $f$ is a continuous function defined over the collection angle of the reflector. The point source is located at the origin $O$. The reflector surface can be written equivalently as an implicit function $F$

$$F(r, \theta, \phi) = r - f(\theta, \phi) = 0. \quad (4.10)$$

The normals to the reflector surface can then be obtained from the gradient of the implicit function $F$ using

$$\mathbf{N} = \nabla F(r, \theta, \phi) = \left( \frac{\partial F}{\partial r}, \frac{\partial F}{\partial \theta}, \frac{\partial F}{\partial \phi} \right). \quad (4.11)$$

By using Eq. 4.10, we have

$$\mathbf{N} = (N_r, N_{\theta}, N_{\phi}) = \left(1, -\frac{\partial f}{\partial \theta}, -\frac{\partial f}{\partial \phi} \right). \quad (4.12)$$

In spherical coordinates, the curl of the field of surface normals is given by

124
\[
(\nabla \times \mathbf{N}) \cdot \hat{r} = \frac{1}{r \sin \theta} \left( \frac{\partial (N_{\theta} \sin \theta)}{\partial \theta} - \frac{\partial N_\theta}{\partial \phi} \right), \\
(\nabla \times \mathbf{N}) \cdot \hat{\theta} = \frac{1}{r} \left( \frac{\partial N_r}{\partial \phi} - \frac{\partial (r N_\phi)}{\partial r} \right), \quad (4.13) \\
(\nabla \times \mathbf{N}) \cdot \hat{\phi} = \frac{1}{r} \left( \frac{\partial (r N_\theta)}{\partial \theta} - \frac{\partial N_r}{\partial \theta} \right).
\]

Using Eq. 4.9 and Eq. 4.12, we can eliminate several terms that we know are equal to zero

\[
\frac{\partial N_r}{\partial \theta} = \frac{\partial N_r}{\partial \phi} = 0, \\
\frac{\partial N_\theta}{\partial r} = \frac{\partial N_\phi}{\partial r} = 0. \quad (4.14)
\]

The only remaining terms in the expression of the curl are

\[
(\nabla \times \mathbf{N}) \cdot \hat{r} = \frac{1}{r \sin \theta} \left( \frac{\partial (N_{\theta} \sin \theta)}{\partial \theta} - \frac{\partial N_\theta}{\partial \phi} \right), \\
(\nabla \times \mathbf{N}) \cdot \hat{\theta} = -N_\phi, \\
(\nabla \times \mathbf{N}) \cdot \hat{\phi} = N_\theta.
\]

The integrability condition in spherical coordinates is therefore

\[
\mathbf{N} \cdot (\nabla \times \mathbf{N}) = \frac{N_r}{r \sin \theta} \left( \frac{\partial (N_{\theta} \sin \theta)}{\partial \theta} - \frac{\partial N_\theta}{\partial \phi} \right) - N_\theta N_\phi + N_\phi N_\theta = 0 \quad (4.16)
\]

\[
\frac{\partial (N_{\theta} \sin \theta)}{\partial \theta} - \frac{\partial N_\theta}{\partial \phi} = 0. \quad (4.17)
\]

If the surface is defined in Cartesian coordinates as \(z = f(x,y)\), we find by following the same steps that the integrability condition is given by
A continuous surface can only fit a field of surface normal if it satisfies the integrability condition given by Eq. 4.17 or Eq. 4.18. An infinite number of source-target mappings may exist, but we must find the subset that also fulfills the integrability condition in order to yield a continuous surface that generates the desired target distribution [83, 84].

How, then, can we design a continuous reflector if the system has no symmetry? Two alternative approaches have so far been proposed. Ries and Muschaweck have proposed a formulation of the problem that is based on standard multigrid numerical integration techniques [20]. However, no details are given on the actual numerical integration technique, which is the key to the difficulty of solving such equations. In Chapter 3 we detailed the Oliker algorithm, which solves the problem for a set of discrete target points using ellipsoids. The major drawback of this approach is that convergence becomes increasingly slow as the number of target points increases. In order to obtain a good approximation of a continuous distribution a large number of target points would be necessary, which is computer-intensive. Interestingly, since the Oliker algorithm provides a discrete solution to the reflector problem, and since we know that as the number of ellipsoids increases the solution converges towards a continuous reflector surface, it also provides a discrete mapping that satisfies the integrability condition. As we shall detail in Section 4.2, we used the Oliker algorithm to generate a mapping that satisfies the integrability condition. This alternate mapping is shown in Figure 4.10a. Similarly to the initial mapping we used shown in Figure 4.10b, all target bins have identical area, and even though the map contours are different, the same overall target distribution is produced.
Since we derived the expression of the integrability condition in spherical coordinates, we can now easily check if a given mapping can yield a continuous reflector or not. In practice, the reflector shape is derived from a set of discrete points, so the integrability condition is calculated from a discrete $\theta$-$\phi$ grid using

$$
\eta = \frac{N_{\phi,i+1,j} \sin \theta_{i+1,j} - N_{\phi,i-1,j} \sin \theta_{i-1,j}}{\theta_{i+1,j} - \theta_{i-1,j}} + \frac{N_{\theta,i,j+1} - N_{\theta,i,j-1}}{\phi_{i,j+1} - \phi_{i,j-1}}
$$

(4.19)

where $N_\theta$ and $N_\phi$ are the components of the surface normals in spherical coordinates, and $i$ and $j$ are the indices of the grid respectively along $\theta$ and $\phi$. Ideally $\eta$ should be zero everywhere if the integrability condition is strictly fulfilled. We calculated $\eta$ for the two mappings shown in Figure 4.10. The two results are shown in Figure 4.11 as a function of the target location. For the traditional mapping using symmetry considerations (on the right), we see that $\eta$ increases.
significantly as we move away from the $x$ and $y$ axes. This is expected, since the $x$-$z$ and $y$-$z$ planes (as well as the diagonal planes) are planes of symmetry of the system. Although it is harder to distinguish in the figure, the diagonals also correspond to a very narrow minimum of $\eta$, which is consistent. On the other hand, the mapping derived from the Oliker algorithm (on the left) minimizes $\eta$ throughout the entire target.

Figure 4.11. Magnitude of $\eta$ for the two mappings presented in Figure 4.10. $\eta$ is plotted as a function of the target map location. The target map on the right results in large residual curl as we move away from the axis. In comparison, the integrable mapping on the left minimizes $\eta$ over the entire target and can therefore be achieved with a smooth continuous reflector surface.

We generated a reflector shape using the mapping that fulfills the integrability condition, using our initial “rib curve” approach shown in Figure 4.6. The resulting target distribution is shown in Figure 4.12. This time the reflector produces the expected uniform square distribution, and its shape is significantly different from the previously generated reflector in Figure 4.6. Some of the green $\phi$ contours on the target map are now curves instead of straight lines. Figure 4.13 validates this observation and shows that the ray fan emitted at $\phi = 22.5^\circ$ forms a line on the reflector but follows a curve on the target in agreement with the integrable map contours.
Figure 4.12. Reflectors generated from the source-target mapping depicted in Fig. 4.5 (using the Oliker algorithm) and their corresponding illuminance distributions.

Figure 4.13. The integrable map for a uniform square target maps a ray fans at various $\phi$ angles onto curves on the target, except along the axis of symmetry of the system.
4.3 Generating maps satisfying the integrability condition

4.3.1 Map generation using the Oliker algorithm

In this section we detail how to retrieve mappings that satisfy the integrability condition, such as the one depicted in Figure 4.10a. The generation process is outlined in Figure 4.14: we first use the Oliker algorithm to obtain a discrete source-target mapping, then we interpolate the discrete data into continuous mapping functions [83].

![Figure 4.14](image_url)

Figure 4.14. Map generation process. (a) Ray directions aimed towards the center of each facet are mapped to their corresponding target point. In order to preserve edges, points at the edge of the target are mapped to the ray directions corresponding to the middle of the outer facets instead of their center. (b) The discrete mapping relationships \( x = f(\theta, \phi) \) and \( y = g(\theta, \phi) \) can then be approximated by a surface to get a continuous mapping function. (c) The generated map plots the contours corresponding to constant \( \theta \) and \( \phi \) values.

Let’s first examine how a discrete source-target mapping can be obtained with the Oliker algorithm (Figure 4.14a). A reflector generated with the Oliker algorithm is made of a collection of ellipsoids that reflects light towards their corresponding target points. In order words, every source ray \((\theta, \phi)\) that hits ellipsoid \(E_i\) is mapped to target point \(T_i\) with coordinates \((x_i, y_i)\). But
since we want to obtain a continuous one-to-one (bijective) mapping between the source and the target, we must associate a single ray direction with each target point. Unfortunately, we cannot know precisely which ray direction corresponds to a given target location unless we use a large number of ellipsoids, so we need to make some assumptions. One possible arbitrary choice is to associate the ray direction aiming towards the center of each facet to its corresponding target point. We also know that the edges of the reflector must correspond to the edges of the target distribution. So for the ellipsoid facets at the edge of the reflector, we use the ray direction corresponding to the middle of the facet outer edge instead of its center as the reference direction, as shown in Figure 4.14a. By following these two rules, we can now associate each target point location \((x_i, y_i)\) with a unique source ray direction \((\theta_i, \phi_i)\), and proceed with the interpolation of the data into continuous mappings.

Special care must be taken to ensure that the mapping is accurate at the edges. Continuous mappings spread the flux originally concentrated at each target point into a smooth target distribution. Figure 4.15 shows this spreading of the flux for a uniform square target. Compared to inner target points, edge target points must only spread light towards the inside of the target (this is because we force the reflector edges to be mapped to the target edges). Therefore, only half of the flux is needed at edge target points, and the prescribed target values for these points should be scaled by 0.5. In the case of a square target, corners present a singularity: only one fourth of the flux is needed compared to inner target points, so the prescribed target values for corner target points should be scaled by 0.25. Without such scaling the generated target maps would exhibit peaks at the edges and at the corners, as shown in Figure 4.15. Alternatively, we
could have chosen not to scale the target edges and to spread flux both inward and outward. In that case reflector edges are mapped to a new location outside the original set of target points. However, this approach appeared to generate edge artifacts that could not be easily avoided.

![Figure 4.15. Scaling of target values for maps.](image)

(a) Flux spread  
(b) Uniform weighting  
(c) With edge scaling

Once an Oliker reflector with appropriate edge scaling has been generated, we can proceed with linking target point locations to source directions. This discrete mapping can then be either interpolated or approximated into a continuous function, as shown in Figure 4.14b. Data fitting is performed in Matlab with the routine `gridfit` [74]. `Gridfit` provides surface approximation of scattered data points using flexible elastic plates. We used it as an alternative to the native Matlab function `griddata`, which only interpolates and cannot handle points that lie outside the convex hull of the data (in other words, it does not extrapolate the data). Additionally `gridfit` provides a parameter that adjusts the smoothness of the gradient of the approximated surface. Other interpolation or approximation techniques can be used to handle scattered data points, such as techniques using radial basis functions [85]. In this case we privileged the speed of the data
fitting routine because we intend to use map generation within iterative algorithms. Reliability of the data approximation can be improved by using implicit knowledge about the mapping. For instance, we know that the edges of the reflector must be mapped to the edges of the target. We can thus add data points along the edges by interpolating existing edge data points, which minimizes possible edge artifacts. The interpolation is performed in “edge space” so that edge singularities are preserved (a regular 3D interpolation of the edges would otherwise round the corners). This is especially useful when the number of data points is small or when the edges of the target have singularities, such as the corners with a square target.

Most data fitting routines, including gridfit, are designed for Cartesian coordinates, and therefore do not handle periodicity occurring with spherical coordinates. Many reflectors however have $\phi$-periodicity, such as the one shown in Figure 4.12. To overcome this limitation, we extend the range of $\phi$ angles beyond $[0,2\pi]$ and replicate existing data points in the extended range, as shown in Figure 4.16. This way we simulate periodicity around $\phi = 0$ and $\phi = 2\pi$. When plotting spherical coordinates over a rectangular grid, the reflector apex corresponding to $\theta = 0$ degenerates into a line (different values are possible for different $\phi$). To prevent this, we force all values at $\theta = 0$ to be identical. Still, the unwrapping of spherical coordinates onto a rectangular grid makes the data points sparse for small values of $\theta$, so the approximation process is more likely to generate errors in that region. One possible solution would be to adapt the approximation routine to directly handle spherical coordinates. We did not follow this path since we felt that optimization of the maps is a more promising approach to solve this problem.
Figure 4.16. Approximation of discrete mapping data with Gridfit. Spherical coordinates are wrapped onto a rectangular grid and the $\phi$ range is extended to emulate periodicity of the data.

4.3.2 Map accuracy

The accuracy of the mappings depends on how finely the target distribution is sampled. In general, uniform and smooth target distributions require a small number of target points, while target distributions with sharp gradients and complex patterns require large number of target points to be rendered correctly. Since we need to run the Oliker algorithm to generate a map, the more target points, the longer it takes, as we saw in Chapter 3. There is a tradeoff between computation time and accuracy. In Figure 4.17 we generated a series of maps for the same uniform square target but with various numbers of target points. The accuracy of the mappings can be quantified by checking the integrability condition given by Eq. 4.19 and by checking that target maps yield the desired flux distribution. For each map we thus plotted the residual projected curl $\eta$ of the surface normals and the deviation between achieved and desired illuminance distributions. Results are shown in Figure 4.17. The illuminance at a given target location is evaluated by computing the area of each bin. Indeed, since the amount of flux is
defined to be the same in each bin, the illuminance (flux per area) can readily be computed by knowing the target area of each bin.

As we increase the number of ellipsoids, the residual projected curl $\eta$ and the bin area error both decrease as expected. In this case a plateau is reached at about 750 ellipsoids where using more ellipsoids brings no improvement to the accuracy of the mapping. The orange dotted curve shows the corresponding computation time required to generate the mappings. In this case 400 ellipsoids yield a good compromise between accuracy and computation time. The spatial distribution of the bin area error and $\eta$ is shown for selected points below each plot. We can see that error distribution is uneven: it is largest close to the target edges, especially in the corners. This is consistent with the fact that target corners are singularities that require high accuracy to be mapped properly.

The residual errors found in the maps come from three considerations:

- We associate each target point to the center of its corresponding ellipsoid facet. This assumption becomes less reliable as the number of ellipsoids decreases. Indeed, there is no guarantee that as we increase the number of target points the location of the facet centers will remain unchanged.

- The accuracy of the reflector generated with the Oliker algorithm depends on the number of rays used during generation and the criteria used to end the algorithm.

- The approximation of the scattered data points into continuous mapping functions is a generic process that does not take into account specific requirements of the map, such as the integrability condition or the equi-area of the target bins.
Figure 4.17. Mapping integrability and flux mapping error for a uniform square target.
While they may contain residual error, such “1st order” maps provide valuable insight and can be generated quickly. As we shall see in the next sections, they can be used to create reflectors and they can be optimized to improve their accuracy.

4.3.3 Some examples of integrable maps

Before moving on to the generation of reflectors from maps, let’s have a look at some target maps for various reflector geometries. Figure 4.18 shows maps for uniform square and rectangular targets. In each case the reflector collects light over a hemisphere. These geometries were chosen because they are common in street lighting applications, but it is worth pointing out that the map retrieval method outlined before can be applied to any target distribution and any point source intensity distribution.

![Target maps for three different system geometries: square, rectangle and off-axis square. In all cases the target is uniform, the point source is isotropic, and the collection angle of the reflector is a hemisphere.](image)

Figure 4.18. Target maps for three different system geometries: square, rectangle and off-axis square. In all cases the target is uniform, the point source is isotropic, and the collection angle of the reflector is a hemisphere.
Maps in Figure 4.18 were generated from initial reflectors made of 400 ellipsoids. The number of ellipsoids was chosen as the result of a tradeoff between the map accuracy and the time it takes to generate the map. In this case, 400 ellipsoids corresponds to a threshold after which increasing the number of ellipsoids has little effect on the maps. Using more ellipsoids would increase the computation time with negligible benefit. In this example all targets have uniform illuminance, which implies that all target cells have the same area. We see that for the square and rectangular targets, the blue iso-θ contours are nearly circular at the center of the target and gradually become squarer closer to the edges of the target. Interestingly, it appears that the rectangular map matches closely a stretched version of the square map. When the target is shifted off-axis (Figure 4.18c), the center of the map is displaced in the opposite direction and the grid pattern changes accordingly: bins close to the source are compressed, while bins away from the source are elongated. But since the target is uniform, all bins maintain the same area.

A good fit for the blue iso-θ contours in Figure 4.18a and Figure 4.18b can be obtained with superellipses (also referred to as Lamé curves). The general equation for a superellipse is

\[ \left| \frac{x}{a} \right|^n + \left| \frac{y}{b} \right|^n = 1 \]

where \( n, a \) and \( b \) are positive numbers. When \( n = 2 \), the curve is an ellipse (or a circle in the special case where \( a = b \)). As \( n \) increases it gradually becomes closer to a rectangle (or a square on the special case where \( a = b \)). This behavior corresponds to the evolution of the concentric contours in Figure 4.18a. To test the validity of this parameterization, we fitted a set of contours using Eq. 4.20. We see in Fig. 4.13 that we obtain a very good fit. As expected the parameter \( n \)
varies between 2 and infinity (so $1/n$ varies between 0 and 0.5). Second order Béziers curves also provide a good fit for the contours.

Figure 4.19. Map contours fitted with Lamé curves. The two Lamé curve parameters $a$ and $n$ are plotted on the right as functions of the source angle $\theta$ ($1/n$ is plotted since $n$ goes to infinity as the contour shape becomes a square).

In order to get a better understanding of the physical meaning of the integrability condition, let’s consider a faceted reflector. In this example we want each facet of the reflector to illuminate the same square area uniformly. In practice, full overlap of the light distribution created by each facet makes the reflector less sensitive to manufacturing tolerances. In Figure 4.20 we show integrable maps for three individual facets centered at $\theta = 60^\circ$ and at various $\phi$ angles. In each case the collection angle of the facet is $20^\circ \times 20^\circ$. Significant changes occur when $\phi$ increases. When $\phi = 0^\circ$ the target map resembles a regular square grid. In fact, because the differences between the actual integrable map and a regular square grid are small, the latter could be used to
generate a reflector shape with minimal effect on the resulting target distribution. On the contrary, when $\phi = 45^\circ$, the map contours rotate along with $\phi$, but in order to fit the square target shape the map contours turn into a diamond shape that appears to bulge at its center.

![Integrable maps for reflector facets with different azimuth angles](image)

Figure 4.20. Integrable maps for reflector facets with different azimuth angles $\phi$ ($0^\circ$, $22.5^\circ$ and $45^\circ$) and a uniform square target. In all cases the elevation angle $\theta$ is $60^\circ$ and the collection angle subtended by each facet is $20^\circ$ by $20^\circ$. 225 ellipsoids were used to generate the maps.

Now let’s imagine that instead of using the integrable maps shown in Fig. 9, we kept using the same fixed regular square map (as in the $\phi = 0^\circ$ case) for all $\phi$ angles. In order to visualize what happens to the rays, let’s imagine that we attach one end of a set of strings to each node in the source grid, and attach their other end to the mapped location on the target grid. The strings simply represent a bundle of rays propagating from the reflector to the target, according to the desired mapping. As we increase $\phi$, the facet starts rotating around the z-axis, but because the target remains fixed the strings start twisting along z. Correspondingly, the required surface normals on the reflector surface must twist to reflect the rays towards their assigned target
location. This twist of the ray bundle is exactly what the integrability condition prevents. The meaning of the curl is to quantify the rotation of a vector field. The integrability condition corresponds to the curl of the surface normals projected onto itself. So the physical interpretation of the integrability condition for reflectors is that whenever a mapping attempts to locally twist the rays along the direction of the surface normal, surface discontinuities must occur. If the shape of the collection angle of the reflector is set so that it matches the shape and orientation of the target, then simple regular maps may closely match the actual integrable map, as in the $\phi = 0^\circ$ case in Fig. 9. Lens arrays, for instance, provide one mapping technique where the lenslets can be rotated so that the integrability condition is satisfied (e.g., [11]).

4.4 Generating smooth reflectors from maps

4.4.1 Motivation

Since the target distribution produced by reflectors generated with the Oliker algorithm is discrete, we must find a way to modify the reflector shape in order to obtain the original continuous target distribution. Ideally, one would simply use a large number of ellipsoids and then interpolate the resulting reflector surface into a smooth continuous surface. However, as we saw in Section 3.2, the computation time scales quadratically with the number of ellipsoids, so it becomes increasingly impractical to use more than a couple of hundreds of target points with our implementation of the algorithm and current computational capabilities. On the other hand, when the number of ellipsoids is small and convergence is fast, there is no easy way to convert the reflector surface made of ellipsoid facets into a smooth surface that will produce the desired continuous target distribution. Figure 4.21 shows illuminance distributions produced by directly
interpolating the initial reflector made of ellipsoid facets with a NURBS surface. We used the same set of surface points that we use to generate maps (one point per facet located at the facet centroid). Direct surface fitting typically produces artifacts in the target distribution such as peaks at the edges, even when the total number of ellipsoids increases. Smoothing the surface must be done in a controlled manner, and generating reflectors from maps provide a good alternative to do so.

Maps provide continuous mapping relationships that satisfy the integrability condition, so we can use them to build smooth reflector surfaces. Figure 4.22 summarizes the full reflector generation process.
1) A reflector is generated from a discrete version of the desired target distribution using the Oliker algorithm.

2) A discrete source-target mapping is retrieved from the reflector, and then approximated into a continuous mapping.

3) A smooth reflector that achieves a continuous target distribution is generated from the continuous source-target mapping.

![Fig 4.22. Generation of a smooth reflector using the Oliker algorithm.](image)

As we shall see over the next sections, there are several additional benefits to generating reflectors from maps. Maps for target distributions with smooth variations can be generated quickly from a small number of target points. Also, maps tend to vary slowly when system parameters are changed (especially when those changes do not involve significant twist in the mapping). This means that the same map can be used to generate reflectors with different system parameters, even though the corresponding reflector shapes may be significantly different.

### 4.4.2 Effect of the integration path

In order to generate a reflector surface from a mapping, one must define the starting point and the integration path. The starting point is arbitrary and sets the overall size of the reflector. Typical choices of starting points are the apex of the reflector or a point at the rim of the reflector.
reflector. The integration path, on the other hand, defines a sequence of positions throughout the map along which integration will be performed. If the integrability condition were fulfilled rigorously, the resulting shape would be independent of the chosen integration path [81]. However, since in our case the integrability condition is not strictly fulfilled, some residual slope error accumulates during integration, and thus some integration paths may yield fewer artifacts in the target distributions than others. The integration path is in general divided into two steps. First, an initial reflector profile is generated by integrating from the starting point along either $\theta$ or $\phi$, using the construction algorithm described in Section 4.1. For each point on this initial strip, integration is then performed in the orthogonal direction in order to cover the collection angle of the reflector. Finally, the set of generated points is interpolated into a smooth continuous surface with a NURBS. An optimal choice of starting point and integration path minimizes the lengths of each path. In the example shown at the top in Figure 4.23, the starting point is set at the apex of the reflector ($\theta = 0^\circ$). From this starting point, we can integrate along $\theta$ (green curve) at a given $\phi$ and then for each point along the initial green curve integrate along $\phi$ (set of blue curves).

Alternatively, one can integrate along $\phi$ first (in this case, since the starting point is the apex, integration along $\phi$ has no effect), and then along $\theta$, which produces a set of green curves that span the entire reflector surface. In this case integrating along $\theta$ first results in a much longer path than integrating along $\phi$ first, so integration along $\phi$ first results in a more accurate target distribution. Indeed, Figure 4.23 shows that integrating along $\theta$ first results in rounded corners and peaks at the corner of the target, while integrating along $\phi$ first yields a very uniform target.
The reflector apex is also a good starting point because the shape tends to be sensitive to mapping errors close to the poles in spherical coordinates.

Figure 4.23. Effect of the integration path on the target illuminance distribution. Longer integrations paths produce more artifacts in the target distribution because residual map errors accumulate during integration, which translates into slope mismatch in the reflector surface.

For reflector facets, the facet center is usually a good starting point and integrating along $\phi$ or $\theta$ first does not make much difference (unless the facet is strongly asymmetric and the angular $\theta$ range of the facet is very different from the $\phi$ range). Figure 4.24 shows the effect of various integration paths for a uniform square target. Once again, longer integration paths (bottom row in Figure 4.24) tend to produce larger errors in the target distribution. In some cases averaging the surfaces generated from various integration paths or various starting points can help minimizing artifacts in the resulting target distribution.
Figure 4.24. Effect of the integration path on the target distribution for a reflector facet. Similarly to the example shown in Figure 4.23, longer integrations paths produce more artifacts in the target distribution because residual map errors accumulate during integration, which translates into slope mismatch in the reflector surface.

4.4.3 *Effect of the number of target points*

When generating reflectors from maps, higher target fidelity can typically be obtained at the expense of computation time. Figure 4.25 shows that increasing the number of target points of the initial reflector improves target uniformity but it increases the time it takes to generate the
In this case a performance plateau is reached at about 600 ellipsoids where increasing further the number of ellipsoids brings no significant uniformity improvement.

Figure 4.25. Effect of the number of ellipsoids in the initial reflector on the target distribution, for a uniform square target. The statistical noise for these simulations is 0.5%. The reflector that produces the uniform target distribution on the right was generated in about 12 s on a 2.5 GHz Intel Quad core computer from an initial reflector made of 441 ellipsoids. The quadrant symmetry of the system was leveraged to speed up computations. The target distribution has 0.7% RMS non-uniformities over the target area, 1.6% peak-to-valley, and the statistical noise inherent to Monte-Carlo ray tracing is about 0.5%. So when target
distributions vary smoothly, smooth reflectors can be accurately generated from a small number of target points, hence very quickly. Target distributions with more complex patterns and high spatial frequencies however will require a larger number of target points and do not benefit from the same gain in speed.

Figure 4.26 shows the sag difference between the shape of the initial reflector made of ellipsoid facets and the smooth reflector surface generated from an integrable map. The system geometry is similar to the example in Figure 4.23 except that the initial reflector is made of 149 ellipsoids. The initial reflector made of ellipsoids is continuous but has slope discontinuities in between ellipsoid facets. On the other hand, the reflector generated from the map is smooth and has neither step nor slope discontinuities. Figure 4.26 illustrates the smoothing action achieved by the map: instead of ellipsoids focusing light towards target points, light spreads out and forms a continuous target distribution. The peaks in the plot correspond to the dips in the original reflector surface where multiple ellipsoid facets meet and slope discontinuities occur. In this case, the largest difference between the original reflector sag and the smooth reflector sag is about 0.8 mm and occurs at the reflector edge.
Figure 4.26. Sag difference between the initial reflector surface made of 149 ellipsoids and the smooth reflector surface generated from its corresponding integrable map. The peaks in the plot correspond to the dips in the original reflector surface where multiple ellipsoid facets meet and slope discontinuities occur.

4.4.4 Examples of smooth reflectors generated from maps

We now present some examples of smooth reflectors generated from maps. In each case we first derived a reflector for a discrete target using the method of supporting ellipsoids, as explained in Chapter 3. We then derived an integrable map and followed the smooth reflector generation method detailed in the previous section. Figure 4.27 corresponds to a typical streetlight configuration. The target is a uniform 60×120 ft rectangle, placed 30 ft away from the source. The point source is isotropic and light is collected over a hemisphere. In this example the map was generated from an initial reflector with 20×20 target points. The target illuminance RMS non-uniformity is about 1.4% and 11% peak-to-valley. 2 million rays were traced in this simulation.
Figure 4.27. Performance of a reflector generated from a uniform rectangular map.

Figure 4.28 is another typical streetlight configuration where the source is located on the side of the illuminated area, so the target is off-axis as seen from the source (shifted along $x$). As we previously saw, the corresponding reflector shape is elongated along $x$, and the target map contours are squashed in the direction opposite to the shift. The resulting target illuminance shows noticeable artifacts, especially close to where the contours are squashed. An initial reflector with more than 400 target points would be required in order to render properly the regions where contour lines are squashed. In this case, RMS non-uniformity is about 2.6% and 30% peak-to-valley. This large peak-to-valley value is caused by the illuminance peaks and dips corresponding to the edge artifacts and it does not reflect the higher level of uniformity at the center of the target.
Target distributions do not have to be uniform. Figure 4.29 shows an example of a square target with a 2D Gaussian illuminance profile (along both $x$ and $y$). 400 target points were used to create the source-target map. If we compare this map with the map for a uniform square target shown in Figure 4.18, we notice that the blue map contours are now more closely spaced in the vicinity of the map center because illuminance is higher at the target center than at the edges. Generating a reflector from the map is also effective in this case. Figure 4.30 plots the deviation between the measured and prescribed target illuminance distributions. The peak deviation is less than 8% and occurs at the corners, where the prescribed target values are the lowest. The average deviation across the entire target is less than 2%, which is in good agreement with the specifications.
Figure 4.29. Performance of a reflector generated from a Gaussian square target map.

Figure 4.30. Target distribution error for the Gaussian square target in Figure 4.29.
Finally, we generated reflector facets from the maps previously shown in Figure 4.20. Each reflector facet illuminates the same uniform square target (facets produce overlapping patterns, which is the reason why faceted reflectors are less sensitive to manufacturing tolerances). In this example the source is isotropic, the collection angle of each facet is a $20^\circ \times 20^\circ$ solid angle, and the source-target maps were generated from initial reflectors made of 225 ellipsoids.

Figure 4.31. Performance of reflector facets generated from uniform square maps.
The reflector facets proved to be more challenging to generate because of singularities at both the target corners and the facet corners. The target map for the facet at $\phi = 0^\circ$ is close to a regular square grid, and the amount of twist in the mapping is small, so the target distribution can be rendered accurately with a map created from only 225 target points. However, the facets at $\phi = 22.5^\circ$ and $\phi = 45^\circ$ exhibit more complex contour variations around the areas corresponding to the facet corners (where the red rays hit the target edge). As a result, the edges of the illuminance distributions present some artifacts around these areas. An initial reflector with a larger number of target points would be required to generate more accurate contours in the vicinity of these sensitive areas of the map.

It is worth highlighting that point sources make the target distribution extremely sensitive to reflector shape variations. Because the reflectors presented in this section achieve one-to-one mappings between the source and the target, any deviation in the source intensity distribution or the shape can have a dramatic impact on the target distribution. We use point sources to test the validity of the reflector generation approach. In practice, extended sources may wash out some of the observed artifacts arising from small errors in the generated shapes.

An alternative solution to improve the accuracy of the reflector facets without increasing dramatically the number of target points is to increase the density of target points in areas where singularities occur. In most cases it means adding target points at the edges of the target or where sharp variations occur in the target distribution. The target values at each of the added target points must then be weighted appropriately in order to maintain proper spread of the flux.
4.5 Evolution of maps with system parameters

The behavior of integrable maps as we change system parameters reveals some interesting properties. In this section we consider the effect of the reflector geometry, the source intensity distribution, the target size, the target aspect ratio and the reflector size on integrable maps. Let’s start with the effect of the reflector geometry. The two reflector shapes shown in Figure 4.32 produce the same uniform square target distribution but use different reflector geometries. The top reflector makes rays cross the z-axis (crossing geometry) while the bottom reflector does not (non-crossing geometry). These two different mappings yield significantly different reflector shapes, however the corresponding target maps appear identical.

![Figure 4.32. Comparison of integrable maps for crossing and non-crossing reflector geometries.](image)

Mappings are therefore not directly correlated with the reflector shape: very different shapes may correspond to very similar mappings. This is convenient, since we can now generate a mapping
for the crossing geometry (which can sometimes be lengthy to compute, as we saw in Chapter 3), and get the non-crossing geometry “for free”, with no additional computation.

When the intensity distribution of the source changes, the source equi-flux grid will change, but the target equi-flux grid may not necessarily change significantly. This phenomenon is evidenced in Figure 4.33, where source-target maps for an isotropic and a Lambertian source are compared.

![Figure 4.33. Comparison of source-target mappings for an isotropic source and a Lambertian source. In both cases the target is a uniform square, and light is collected over a 120° cone.](image)

The source grid for the isotropic source on the left has unevenly spaced blue contours. Indeed, the bins at the pole must be wider than at the rim in order to have the same solid angle, and thus collect the same amount of flux. The Lambertian emission pattern emits more light towards the pole than towards the rim, therefore the bins close to the pole are smaller than at the rim, and the blue contours end up being more evenly spaced. While the source grids are significantly different for the isotropic and Lambertian sources, the target grid appear identical, so in this case new target grids do not need to be computed when the source distribution changes.
Figure 4.34 shows the effect of the target size on the maps. A light source placed at 30 ft above a street illuminates uniformly a square area. The size of the illuminated area varies from 30×30 ft to 240×240 ft. In all cases we use an isotropic point source and the collection angle of the reflector is a hemisphere. Because the corresponding changes in the maps are small, we plotted map variations as the displacement of the nodes of the target maps relative to the 30×30 ft map, with a magnification factor of 4.

Each arrow in Figure 4.34 thus corresponds to the displacement of a node of the reference map shown on the left. These displacements are a result of the need to fulfill the integrability condition as the target size increases. The largest map variations occur close to the edges of the
target. When the target size increases from 30×30 ft to 240×240 ft, the maximum node displacement over the entire map is less than 4% of the target width. Map variations thus remain small when the target size is changed, which means that we can generate reflectors from a given map for target sizes different from the original target size with minor effects on the resulting target distribution. The corresponding reflector shapes, however, may vary significantly, as can be seen in Figure 4.34.

We now study the effect of the aspect ratio of the target on the target map. The initial target is a uniform 60×60 ft square and the target is gradually stretched into a uniform 240×60 ft rectangle. In Figure 4.35 we show the displacement of the nodes of the target map (with ×2 magnification) as the aspect ratio of the target increases. In each case we scaled the rectangular target map back to a square target map, and then calculated the node displacement relative to the reference square target map on the left. The goal here is to investigate how close rectangular maps are to a stretched version of the original square map. It appears that maps for uniform rectangles with various aspect ratios correspond closely to stretched versions of the uniform square map. We see that even when the target width is doubled (2:1 aspect ratio), the displacement of the nodes remain small; the maximum node displacement, which occurs at the edge of the target, is less than 2% of the target height. When the target size is quadrupled (4:1 aspect ratio), the maximum node displacement becomes about 4.3% of the target height. So it is possible to generate reflectors for different target ratios from a single square map, hence saving computation time. As Figure 4.35 shows, the corresponding reflector shapes may be significantly different, ranging from a quasi-circular aperture to a football ball shape.
Figure 4.35. Effect of the target aspect ratio on the target map. The target is located at $z = 30$ ft, and its size varies from 60×60 ft to 240×60 ft. The point source is isotropic and light is collected over a hemisphere. All reflectors are about 1 ft in diameter.

Finally, we studied the effect of the reflector scale on the target map. We used the same typical setup as in the previous examples with an isotropic source located 30 ft away from a 60×60 ft target. The target is uniform and the collection angle of the reflector is a hemisphere. We show in Figure 4.36 four reflectors with output diameters spanning 3 orders of magnitude: 0.1 ft, 1 ft, 10 ft and 100 ft (which might be a bit too large for any practical purpose, but the goal here is to understand the effect of the size of the reflector relative to the target). We use the 1 ft diameter reflector as the reference target map. As we can see, the target map in this case is extremely insensitive to the reflector scale. For reflectors 10 times smaller or larger than the reference 1ft reflector, node displacements are barely noticeable, even when magnified 5 times.
Quantitatively, the maximum node displacement for the 10 ft reflector corresponds to only 0.25% of the target height (average displacement is about 0.08%), while the maximum node displacement for the 0.1 ft reflector is less than 0.06% of the target height. This means that reflectors can easily be scaled without having to compute a new target map. This is especially useful since it can be difficult to choose precisely the scale of the reflector with the Oliker algorithm, as we discussed in Section 3.2.1.

![Figure 4.36. Effect of the reflector size on the target map. The map variations are plotted as a displacement of the map nodes relative to the reference map of the 1 ft diameter reflector. The target is a uniform 60x60 ft square and is located at z = 30 ft. The point source is isotropic and light is collected over a hemisphere. The output diameter of the reflector varies from 0.1 ft to 100 ft. Each map thus corresponds to an order of magnitude difference in reflector scale compared with its adjacent maps.](image)
CHAPTER FIVE - EXTENDED SOURCES

This chapter investigates the effects of extended sources as opposed to designs based on the point source approximation. Section 5.1 first introduces the most common effects of extended sources on the target distribution and highlights the limitations imposed by systems using a single reflective surface. Section 5.2 details a design method for rotationally symmetric reflectors that iteratively adjusts the prescribed target distribution in order to compensate for the measured effects of the source extent. Finally, Section 5.3 proposes a method to design freeform reflectors with extended sources based on tiling the source images produced by a reflector generated with the Oliker algorithm.

5.1 Understanding the effects of extended sources

A point source has no spatial extent and therefore is fully characterized by its intensity distribution. On the other hand, extended sources are characterized by their luminance distribution, which means that both spatial and angular emission properties must be taken into account. For Lambertian sources, the luminance is constant, and the intensity emitted in a given direction is directly proportional to the projected area of the source in that direction. Figure 5.1 shows the correlation between the apparent area of the source and its intensity for an LED. Even when the source is not perfectly Lambertian, projected area often remains strongly correlated to the source intensity distribution.
Figure 5.1. Intensity distribution of an LED and apparent source area in several directions. For a Lambertian source, the larger the projected area of the source, the higher the intensity.

In Chapter 2 we gave a brief overview of the effects of extended sources. Monte-Carlo ray tracing can be used to evaluate the target distribution, usually based on a simplified model of the source or on measured data. The most commonly observed effects are smearing of the edges of the target distribution and either peaks or dips at its center. This behavior can be evidenced by looking at the pinhole camera images of the source created by the reflector, as shown in Figure 5.2. Each point of the reflector can be thought of as creating its own image of the source on the target. The orientation and the size of the source image depend on the apparent size of the source as seen from a point on the reflector surface. If the angle subtended by the source as seen from a point on the reflector is large, then a large source image will be created on the target. Conversely, a point on the reflector further away from the source will yield a smaller source image. The absolute size of the source does not matter, and an LED is not any closer to the point source
approximation than any other type of the source. It all depends on the apparent size of the source as seen from the reflector surface. Since designs often have compactness constraints, most designs are scaled to a level where the effects of the source extent become noticeable. In the end, the final target distribution created by the reflector corresponds to the convolution of the target distribution produced by a point source with the varying source image kernel produced by the extended source across the target area.

![Pinhole camera images of the source produced by various points on the reflector.](image)

Any arbitrary target distribution can be obtained with a point source and a single reflector surface. This is not so true with an extended source. Single surfaces do not provide enough degrees of freedom to fully control the target distribution. Let’s consider the example shown in Figure 5.3, where a cylinder source is used to create a bright spot on the target. In order to do so we focus all the rays emitted from the centroid of the source to the center of the target. The reflector shape is thus an ellipsoid, where the centroid of the source is placed at one of the ellipsoid foci and the target center is placed at the second focus. However, once the mapping of
the rays has been set, the shape of the reflector is fully defined and there are no degrees of freedom left to control the orientation and the relative size of the source images.

Figure 5.3. Limitations of single reflectors with extended sources. In this example, all rays coming from the centroid of the source are focused to a point at the center of the target. The corresponding reflector shape is an ellipsoid. Once the mapping has been set, no degrees of freedom are left to control the size and orientation of the source images.

The smearing pattern produced by the superposition of the source images in Figure 5.3 cannot be controlled as long as we use a single surface and force all the rays to aim towards the target center. At best, the scale of the reflector can be changed to either increase or decrease the size of the spot. A similar phenomenon happens when a sharp cutoff is required at the edge of the target distribution. One way to obtain a sharp cutoff is to aim source edge rays towards the edge of the target. But again, once the mapping of the rays is set, a significant portion of the reflector shape is defined, and not enough degrees of freedom are left to control the center of the target distribution. As we shall see, the level of control over the target distribution depends primarily on the relative size of the source compared to the reflector.
5.2 Compensation method for systems with rotational symmetry

5.2.1 Overview of the iterative compensation algorithm

The first design method for extended sources we propose is intended for systems with rotational symmetry. We saw in the previous chapters that the jump in complexity from rotationally symmetric reflectors to freeform reflectors is significant. Therefore, it seems reasonable to start tackling the challenge of extended sources by considering first systems with rotational and translational symmetry. In this case the design principles within the point source approximation are well established and computing the reflector shape takes less than a second. The compensation approach we propose leverages these benefits by integrating point source tailoring within an iterative algorithm that applies a correction factor to the prescribed target distribution based on the extended source effects. The idea of using point source tailoring within an iterative algorithm has been previously explored by Bortz and Shatz [64]. An iterative method using tailored edge rays was also presented in [63]. In our approach, we used the source location and the maximum extent of the prescribed output distribution as additional optimization variables, and we studied over which range of source sizes the algorithm is effective.

Someone familiar with the design of optical imaging systems might wonder why we do not use traditional optimization techniques in a similar fashion for non-imaging systems. We outlined some of the fundamental differences between imaging and non-imaging systems in Section 1.2, but it is worth detailing further the implications of these differences on optimization here. The metrics to evaluate the quality of an imaging system can be, among others, the RMS spot size, the wavefront error, or the values of the aberration coefficients. In all cases these metrics are
directly related to image quality and they can be evaluated by tracing a couple dozens of rays throughout the system. Moreover, because the sequence of surfaces intersected by the rays is known, propagating the rays through the system can be computed very efficiently. Non-imaging systems, on the other hand, are *non-sequential*, which means that any entity in the system must be tested in order to find ray intersections. More computations are thus required for each ray, and at least 1,000 times more rays are typically required to evaluate the target distribution (depending on the level of accuracy desired, several millions of rays may need to be traced). Finally, the complex reflector shapes we have been designing require dozens, sometimes hundreds of variables to be rendered accurately by an analytical surface. In the end, the combination of more variables, more rays, and more computations per ray makes a non-imaging optimization problem at least $10^6$ times more computationally intensive than a typical imaging optimization problem. Additionally, merit functions based on Monte-Carlo ray tracing produce noise, and local optimization of a reflector surface often exhibit local minima where local optimization techniques can easily get stuck. So the use of a local optimizer that modifies the reflector shape iteratively is not an efficient method, even though it may soon be within the reach of computational capabilities on standard computers. The compensation method detailed in this section does not directly modify the reflector shape, but instead applies a correction factor to the prescribed target distribution. The only lengthy operation in the process is the evaluation of the target distribution created by the reflector with the extended source.

We first need to define the terminology used in this chapter. Our goal is to obtain a given intensity or illuminance distribution, which we call the target distribution, or $E_{\text{target}}(x)$, $x$ being
either an angular or spatial coordinate. Since we are dealing with rotationally symmetric systems we only need to control the target distribution along a single 1D profile. The prescribed target distribution that we will use to generate the reflector shape is denoted $E_{\text{prescribed}}(x)$. The reflector shape is generated using the traditional point source tailoring techniques detailed in Section 2.1. Finally, we evaluate the actual target distribution produced by the reflector in simulation software, and we call it $E_{\text{measured}}(x)$. For a point source, we necessarily get $E_{\text{measured}} = E_{\text{target}}$ when $E_{\text{prescribed}} = E_{\text{target}}$ as this is what the point source tailoring equations were derived for. However, for a finite source size $E_{\text{measured}} \neq E_{\text{target}}$ when $E_{\text{prescribed}} = E_{\text{target}}$. In order to compensate for the departure from the target distribution, we apply a correction factor to the values in the prescribed distribution, according to the flowchart shown in Figure 5.4. We then re-evaluate $E_{\text{measured}}$ and we iterate until the measured distribution is close enough to the target distribution.

![Figure 5.4. Flowchart of the iterative compensation algorithm.](image)

The merit function (MF) quantifies the departure of the measured distribution from the desired target distribution. We define the merit function as

$$MF = \sum_i w_i^2 \left( E_{\text{target}}(x_i) - E_{\text{measured}}(x_i) \right)^2,$$

(5.1)
where $w_i$ are the weights for each point $i$ along $x$ where the intensity or illuminance distribution is measured. Methods to efficiently evaluate the target distribution will be detailed in the next section.

Throughout this chapter we always consider normalized intensity and illuminance distributions: only the relative shape of the target distribution matters. Achieving a given intensity or illuminance value is just a matter of scaling the total power of the source, the shape of the distribution remaining unchanged. In some cases only a subset of the output distribution is specified. For instance, we might want to obtain a uniform central distribution no matter what the falloff is at the edges. In these cases, we only normalize the portion of interest in the output distribution. Another merit function item can then be added to weight the flux contained in the portion of interest relative to the total flux (i.e. the efficiency of the system).

### 5.2.2 Target distribution evaluation

The output distribution is evaluated at a set of discrete points over the target extent. Ideally, the target distribution should be evaluated at the same locations where the target values are prescribed so that a direct correlation can be made between prescribed and measured values. Otherwise, the measured values are interpolated at the locations of interest. We evaluate the value at each target point using Monte Carlo ray tracing. The total number of target points and their location can be adjusted depending on the accuracy needs. Importantly, the number of rays traced during the merit function evaluation must be set so that statistical noise inherent to Monte Carlo ray tracing does not become a limiting factor (see Appendix B.1 for more details on the statistical noise associated with Monte Carlo ray tracing). In order to minimize the statistical
noise, the output distribution is averaged for rotational symmetry. As we can see in Figure 5.5, the circular target area is divided into rings and target values are calculated based on the number of rays in each ring. The size of the rings must be chosen according to a tradeoff between resolution and accuracy. Target points can be spread out further from each other closer to the pole, so that the number of rays in each slice is more likely to be comparable, ensuring the same statistical noise level across the target (equi-solid angle slices at the bottom of Figure 5.5). However, having a lower point density close to the center makes it difficult to control accurately the target distribution in that region. For that reason, we decided to use an evenly spaced grid of target points, which thus results in fewer rays in center slices and noisier measured target values at the center of the distribution than at the edge (equi-angle slices at the top of Figure 5.5).

Figure 5.5. Illustration of the tradeoff between the measured target point density and the statistical noise inherent to Monte Carlo ray tracing.
5.2.3 Adjusting the target distribution extent

The compensation method relies on the point source tailoring technique we reviewed in Section 2.1 to generate the reflector shape. The required parameters to generate the shape are the intensity distribution of the source, the collection angle of the reflector, the prescribed target distribution, and the reflector scale (typically the polar radius at either the apex or the rim of the reflector). Among these parameters, the source intensity distribution and the collection angle of the reflector are often specified. The reflector scale is driven by compactness requirements, so this parameter may not necessarily be available as a degree of freedom to the designer, although it has a tremendous impact on the target distribution: the larger the reflector, the smaller the source relative to the reflector, and thus the closer we are to the point source approximation, making the effects of the source extent less noticeable. Unfortunately, compactness is often desired and increasing the reflector size to solve our problem is usually not an option. We are left with the target distribution as the primary degree of freedom for our compensation algorithm. In practice, we specify the prescribed distribution with a discrete set of points, and a continuous distribution can be obtained by interpolation of the specified target values.

Let’s have a look at a typical starting point for the compensation method. For the first iteration of the algorithm the prescribed target distribution is set equal to the desired target distribution. Figure 5.6 shows the deviation between the desired target distribution (black curve) and the measured target distribution (red curve), which exhibit typical extended source effects. We face right away a problem: prescribed target values that are set to zero (when the output angle is over 20º) yield non-zero measured target values. In other words, we get flux where the prescribed
target value specifies that there should be no flux at all. This is an issue, because we cannot prescribe target values less than zero!

![Graph showing relative output intensity vs. output angle](image)

Figure 5.6. The extent of the prescribed distribution can be adjusted so that the falloff of the measured output distribution matches the falloff of the target distribution.

This type of situation often happens at the edge of the target distribution right from start, but as we will see in the next section the algorithm can run into situations where the algorithm “gets stuck” trying to remove flux where the corresponding prescribed target value is already zero. It occurs because the extent of the source images spills flux beyond the intended target location. To avoid this, we can adjust the extent of the prescribed distribution so that the falloff of the measured output distribution matches the desired target falloff. In the example shown in Figure 5.6, we reduced the maximum output angle from 22° to 18.5° so that the tails of the target and measured distributions match (blue curve). It is therefore important to set the extent of the prescribed distribution as a degree of freedom for the compensation algorithm. Increasing the
size of the reflector, whenever possible, is also a good way to reduce the effect of the source extent, by minimizing the extent of each source image.

The position of the point source can be used as an additional degree of freedom. As we use a point source to model an extended source, we need to find where to judiciously place the point source relatively to the real source. Using the centroid of the extended source is one possible solution but it does not account for the specific emission characteristics of the source. Another possibility is to use the highest luminance spot of the source as the source location for each emission direction. In this case, the source location moves as the source angle \( \theta \) increases. Elmer recommends a “Phantom source” location in order to minimize the smearing produced by extended sources [15]. The Phantom source concept can be applied in a more general sense. When the source location is set as a degree of freedom, it can be leveraged to shift the location and the size of the images of the source on the target, and can thus be useful to control peaks or dips in the output distribution.

5.2.4 Calculating the correction factor

Once the extent of the target distribution has been scaled properly, the optimization algorithm applies a correction factor to the current prescribed distribution. Alternatively, a new prescribed distribution can be calculated using a simple compensation formula such as

\[
E_{\text{measured}} \neq 0, \quad E_{\text{prescribed\ new}} = E_{\text{prescribed\ old}} \left( \frac{E_{\text{target}}}{E_{\text{measured}}} \right)^{\alpha},
\]

(5.2)

where \( \alpha \) is a power factor that can be adjusted to control the strength of the compensation factor.
If $E_{\text{measured}} = 0$ or $E_{\text{prescribed new}} > E_{\text{max}}$, then $E_{\text{prescribed new}} = E_{\text{max}}$. This maximum allowed prescribed value prevents the algorithm from setting very large prescribed target values when a measured value is close to zero.

As opposed to the case where the measured distribution is non-zero when the prescribed distribution is zero, there are also cases where increasing the prescribed distribution does not increase the measured distribution. In these cases the compensation algorithm may enter in a positive feedback loop where some specific target values become disproportionately larger than the rest of the distribution. Setting a maximum allowed prescribed target value prevents the algorithm from getting in such a loop. Whenever this situation happens, it is usually a sign that the extent of the target distribution needs to be changed, or that the size of the reflector must be increased (the compensation approach becomes less effective as the apparent size of the source increases, as we shall see in the next section).

In the end, changing the prescribed target distribution is equivalent to changing the source-target mapping. When the prescribed target values increase then the density of rays in the corresponding target region must increase, as shown in Figure 5.7. Conversely, a decrease in target value corresponds to a lower ray density on the target. The source-target mappings generated by comparing the cumulative flux distributions of the source and the target are one-to-one (bijective) maps. When a target value goes towards infinity, rays on the target tend to focus to a point and create a caustic. On the other hand, when a target value goes toward zero, rays on the target spread out and a slope discontinuity occurs in the reflector shape (rays “skip” an area of the target). This phenomenon tends to make the prescribed target values behave in a non-
linear fashion. To avoid this nonlinearity, the location of the rays on the target could be directly used as a degree of freedom instead of using the prescribed target values. Recently, Cassarly proposed a cumulative flux compensation approach that avoids most of the undesirable behaviors observed when directly compensating the prescribed target distribution [86].

![Figure 5.7. Correspondance between the density of the rays on the target and the prescribed target distribution. Regions with high prescribed illuminance correspond to high ray densities.](image)

5.2.5 Case study: LED flashlight

In this example, we consider a flat Lambertian emitter. Both the reflector and the source are rotationally symmetric. As a constraint, we set the output diameter of the reflector to be 9.5 mm. The geometry of the system is showed in Figure 5.8. As we can see, each half of the reflector illuminates the full output distribution (from $\beta_{\text{max}}$ to $\beta_{\text{min}} = -\beta_{\text{max}}$). The upper and lower halves of the reflector thus produce identical intensity patterns that overlap. This choice of reflector geometry lowers the sensitivity to the source placement and to the variations in its luminance distribution. We want to study the flexibility of the method for various types of light distributions and source sizes.
Figure 5.8. System geometry of the LED flashlight case study. The system is made of a flat emitter and a rotationally symmetric reflector. As a constraint, the aperture diameter must be equal to 9.5 mm. In this case $\alpha_{\text{min}} = 90^\circ$, $\alpha_{\text{max}} = 150^\circ$, $\beta_{\text{max}} = 22^\circ$, and $\beta_{\text{min}} = -22^\circ$.

We first consider a Gaussian target intensity distribution. The FWHM of the distribution is 16º. The point source has a Lambertian intensity distribution and the reflector output diameter is set to 9.5 mm. We only consider the rays that hit the reflector, which is assumed to be specular. Direct lighting and surface scattering are not taken into account at this point. 100,000 rays are traced during each merit function evaluation. Figure 5.9 shows the effect of the extent of the source as its size increases. The most noticeable characteristics are an increasing dip at the center and smearing at the edges, which eventually lead to a flattened central distribution between 2 mm and 4 mm diameter source. Interestingly, while smearing remains marginal when the source diameter is smaller than 2 mm, on-axis intensity decreases by as much as 45%.

Because the source and the target have the same geometries, there is no limitation due to skewness conservation in this case. The étendue of the source is 1.85 mm$^2$·sr when the source diameter is 1 mm (we only take into account the light collected by the reflector). As a constraint the diameter of the output aperture of the reflector must be 9.5 mm.
Figure 5.9. Measured output intensity distribution as the diameter of the source increases. The table on the right indicates the maximum apparent source size as seen from the reflector in each case, as well as the half-angle of the output distribution in the étendue-limited case (the étendue limit is calculated for the light collected by the reflector).

According to étendue conservation, the minimum half-angle of the cone in which light can be collimated is about 6°, which is well below the total extent of the target. However for the 4 mm source diameter the apparent source size becomes about half the total extent of the target, which violates the point source approximation. In general, designs close to the étendue limit are more easily handled with methods based on edge rays. We can gain some insight on the system by looking at the apparent size of the source as seen from different points on the reflector. In Figure 5.10, the apparent size of the source is plotted against the output angle (the opposite half of the reflector creates a symmetric curve). The largest apparent source size occurs around the center of the target (\(\beta = 0\)), which is the reason why a dip is observed. In order to compensate for this effect, the shape of the reflector must be altered so that the images of the source on the target are shifted towards the center of the target.

<table>
<thead>
<tr>
<th>Source diameter</th>
<th>Max apparent source size</th>
<th>Étendue-limited output half-angle</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.25 mm</td>
<td>1.0°</td>
<td>1.3°</td>
</tr>
<tr>
<td>0.5 mm</td>
<td>2.1°</td>
<td>2.6°</td>
</tr>
<tr>
<td>1 mm</td>
<td>4.1°</td>
<td>5.2°</td>
</tr>
<tr>
<td>2 mm</td>
<td>8.3°</td>
<td>10.5°</td>
</tr>
<tr>
<td>4 mm</td>
<td>17.3°</td>
<td>21.4°</td>
</tr>
</tbody>
</table>
Figure 5.10. Apparent source size in degrees as seen from the reflector for various output angles.

Figure 5.11 shows the evolution of the merit function value over 25 iterations for each source size. It takes about 10 to 20 iterations to reach the minimum of the merit function for the various source diameters.

Figure 5.11. Evolution of the MF values during the iterative process. In all cases it takes about 20 cycles to reach a minimum.
These simulations were performed on an Intel dual-core 2.5 GHz laptop and 100,000 rays were traced for each iteration. The total time required to perform 20 iterations with this configuration is about 2 minutes. Although other approaches to compute the intensity distribution may be used for some cases, Monte-Carlo raytracing was used because it can be applied to a wide variety of different source models. The choice of the correction factor in the iterative algorithm can also improve the convergence time of the algorithm, but usually it has little impact on the final merit function value.

The results presented in Figure 5.11, Figure 5.12 and Table 5.1 show that the iterative algorithm can achieve the desired target when the maximum apparent size of the source is below 1/10\textsuperscript{th} of the full target size. It usually starts failing when the maximum apparent size of the source is between 1/10\textsuperscript{th} and 1/5\textsuperscript{th} of the full target size. The standard deviation of the departure from the target (Table 5.1) increases by a factor of 3.6 when the source diameter doubles from 0.5 mm to 1 mm, but it only increases by a factor of 2 when the source diameter increases from 1 mm to 2 mm. It is also worth noting that as the source becomes smaller, the output distribution becomes much more sensitive to the reflector shape, especially in the region where the rays are aimed at the center of the distribution. The subsequent use of a traditional optimization algorithm can help improve the results.

The initial and final shapes of the reflector are compared in Figure 5.13 for the 1 mm source diameter case. The maximum deviation is about 160 μm.
Figure 5.12. (Left) Measured intensity distributions and (Right) corresponding prescribed intensity distributions after 25 iterations of the compensation algorithm.

Table 5.1. Standard deviation of the departure of the final measured intensity distribution from the target. As the source diameter increases, it becomes more difficult to closely match the target.

<table>
<thead>
<tr>
<th>Source diameter</th>
<th>( \sigma_{\text{error}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5 mm</td>
<td>0.0035</td>
</tr>
<tr>
<td>1 mm</td>
<td>0.0127</td>
</tr>
<tr>
<td>2 mm</td>
<td>0.0237</td>
</tr>
</tbody>
</table>

Figure 5.13. (Left) Comparison of the shapes of the reflector before and after the iterative process for a 1 mm diameter source. (Right) For each Z coordinate the difference in Y coordinates of the initial and final reflector is plotted.
The point source approximation assumes a one-to-one correspondence between the source and the output distribution. Each ray emitted by the point source hits a unique point on the target. With an extended source, each point on the reflector creates an image of the source on the target. The final output distribution can be seen as the superimposition of all the images of the source from each point on the reflector. If the size of the image of the source at any given point on the reflector is comparable to the extent of the desired distribution, then the one-to-one correspondence principle is broken and using the point source approximation is not likely to succeed. However, even when the source size has a significant impact on the output distribution, there is a range of source sizes for which the iterative algorithm is effective. Typically, the method can be effective as long as the source images do not cover more than one fifth of the full target extent.

If we change the characteristics of the source so it becomes non-Lambertian, we can simply reiterate the process in order to compensate for the effect of the new source intensity pattern. In Figure 5.14 the original Lambertian emitter was replaced by a side-emitter, which created smearing and a dip in the center of the output distribution (20% drop). After 10 iterations, we obtain an output distribution close to the original result (the source intensity distribution can equivalently be used as a variable instead of the output distribution). It is worth noting that in this case a drastic change in the source intensity pattern has small effects on the output distribution. This is because the overlap of the two symmetric output distributions created by each half of the reflector largely compensates for the source variations.
Another classical example of output distribution is a uniform distribution with a sharp cutoff. We know that this type of distribution cannot be achieved with a single reflector. We want to determine how our algorithm handles cases where only subsets of the output distribution need to be tailored. In this example we use a 1 mm diameter Lambertian emitter. We want to obtain a uniform intensity over a 10° half-cone; the intensity falloff beyond 10° is of lesser importance. Figure 5.15 shows the results. We obtain a uniform output after about 20 iterations. In this case, we increased the maximum output angle of the prescribed distribution to prevent the falloff observed at 9° in the initial output distribution.
Figure 5.15. (Left) Measured output intensity distributions and (Right) prescribed distribution before and after the iterative process. In this case we use a 1 mm diameter source and the target is a uniform intensity distribution up to 10°.

The iterative method presented in this paper proves to be a simple and effective design technique when the apparent size of the source as seen from the reflector is no more than 1/5th the full extent of the target distribution. The main advantages of this method are its speed and flexibility. Source non-idealities can easily be taken into account. The convergence time depends on the geometry, the correction function used, and the accuracy required; in general it take at most 30 iterations for the merit function to reach a minimum, which is a matter of minutes on an ordinary computer according to today’s standards. The method can also be used as a first step towards further optimization since local optimization algorithms tend to perform better when started close to a solution. Alternative parameterization schemes can also be effective. For instance, one can use the location of the rays at the output as the optimization variables rather than the prescribed intensity.
5.3 Tiling method for systems without symmetry

The previous section covered a design method for extended sources and systems with rotational symmetry. In this section we propose a different method that suits systems with no symmetry.

5.3.1 Controlling the illuminance of the source images

Let’s first have a look at the effects of the source extent with a freeform reflector generated with the Oliker algorithm. The reflector shown in Figure 5.16 was generated from a uniform 10×10 target grid. The target distribution produced by an isotropic point source is shown on the left. Since we prescribed a uniform target, we obtain a matrix of points with equal flux. Let’s now replace the point source by a more realistic cylindrical surface source (Figure 5.16c). Each ellipsoid patch creates its own image of the source. The shape, size, and orientation of the source images depend on the location of the ellipsoid patches. Patches close to the center of the reflector “see” the base of the cylindrical source, thus creating a circular source image. Patches close to the reflector rim see the full length of the cylindrical source, thus creating an elongated source image. Since the flux collected by each ellipsoid patch is about the same but the size of the source images changes, the average illuminance of the source images also changes. As shown in Figure 5.16c the average illuminance of the source images on the outer target is lower than at the center, where source images are smaller. As we shall see, we can modify our original shape generation algorithm to take into account the illuminance at each target point instead of the flux collected by each ellipsoid patch.
Figure 5.16. (a) Single convergent reflector obtained with a 60×60 ft uniform target grid placed 30 ft away from the reflector. 10×10 target points were used. Target illuminance (b) with a point source, (c) with a cylindrical source and the standard algorithm, (d) with a cylindrical source and the PSA-based algorithm. All illuminance maps are normalized by the peak illuminance. 200×200 bins were used and 2.5 million rays were traced.

**PSA-based algorithm**

Illuminance at a given target point can be computed using the projected solid angle of the illuminating beam at that point. If the source is Lambertian with luminance $L_0$, then the reflector acts as a secondary Lambertian source (luminance is conserved if we assume specular reflection). In order to know the illuminance at a given target point, we need to figure out which parts of the reflector illuminate that point. This is often referred to as the *flashed area* of the reflector. The illuminance $E$ at a target point is proportional to the projected solid angle $\Omega$ subtended by the flashed area as seen from that target point, and is given by

$$E = L_0 \iint \cos \theta \sin \theta \, d\theta \, d\phi = L_0 \iint d\Omega = L_0 \Omega. \quad (5.3)$$

The value of the projected solid angle (PSA) can then be calculated from the projected area of the intersection between a unit sphere and the beam, as explained in Appendix A. In order for the shape generation algorithm to converge, we need to impose a restriction: we do not consider the
possible overlap between source images. In other words, we only measure the illuminance at $T_i$ of the source image produced by ellipsoid $E_i$. This is because the algorithm relies on the fact that scaling ellipsoid $E_i$ should only change the value measured at $T_i$. When this condition is not fulfilled, then the algorithm can diverge. In fact, this assumption greatly simplifies the computation of the projected solid angle: we simply need to compute the projected solid angle subtended by the ellipsoid patch $E_i$ as seen from target point $T_i$, as shown in Figure 5.17. If we were taking the overlap of the source images into account, the flashed area would be discontinuous and scattered over multiple ellipsoid patches, which would make a quick evaluation of $\Omega$ more difficult.

![Figure 5.17. Illuminance at a given target point can be computed from the projected solid angle of the beam illuminating that point. When sources images do not overlap, each target point is simply illuminated by its corresponding ellipsoid patch.](image)

An example of the distribution obtained when using the PSA instead of the flux is given in Figure 5.16d. We see that all source images now have the same illuminance, which is consistent with the uniform prescribed target values. While illuminance is not constant over the source
images (the illuminance value derived from the projected solid angle is only valid at the target point location), this is usually a reasonable 1st order approximation for the purpose of establishing a starting point that will then be improved via optimization.

**Compensation method**

In the previous section the compensation method we described for rotationally symmetric reflectors uses prescribed target values as optimization variables. The same approach can be used for the non-rotationally symmetric case. The PSA-based algorithm we just described is useful to control the illuminance at each target point. However, it assumes that the source is Lambertian and that the source images do not overlap. Non-Lambertian emission patterns, overlap between source images, and scattering can create significant deviations between the prescribed and measured illuminance values. Whenever a deviation is measured, we can apply a correction factor to the original prescribed values. When the size of the source images is much smaller than the target, this type of iterative approach is efficient. Convergence is not guaranteed with larger source images, since each source image, depending on its size, may have an impact on many other target values at once.

**5.3.2 Scaling source images**

While any prescribed distribution can be obtained with *theoretical* point sources, this is not so true with *real* extended sources. Inherent limitations relative to single reflectors have been pointed out for rotationally symmetric systems in Section 5.1. Similar observations are still valid for reflectors without symmetry. For instance, enforcing a sharp cutoff at the edges imposes a
constraint on the reflector shape that does not leave enough degrees of freedom to fully control the inner illuminance distribution. Likewise, neither the source image magnification nor the source image orientation can be arbitrarily controlled with a single surface. For given system specifications, the existence of a solution is difficult to establish in general. However, first-order considerations can help understanding if a design is likely to be feasible or not. The shape generation algorithm uses a scale parameter that defines the relative size of the reflector compared to the source. This degree of freedom has a tremendous impact on the target distribution. Figure 5.18 shows various target illuminance distributions obtained when we increase the size of a Lambertian cylindrical surface source of length $L$ and aspect ratio 2:1 (note that the reflector and target sizes are fixed). As the source size increases, so does the size of the source images on the target. For example, when $L$ is equal to 2 mm, the source images under-fill the target and gaps between the source images are evident. With a moderate size source (e.g., $L$ equal to 3.6 mm and $L$ equal to 5 mm), the source images partially overlap. When $L$ is larger than 10 mm, the source images show significant overlap. The size of the reflector can be scaled up or down to control the size of the images relative to the spacing between target grid points.

![Figure 5.18. Evolution of the target illuminance as the size of the source increases. Ideally, the target grid and the reflector shape should be adjusted so that source images tile the target area. 120×120 bins were used and 2.5 million rays were traced.](image)
5.3.3 *Tiling source images*

Designs are typically started with a uniformly spaced target grid, as shown in Figure 5.16. After a first reflector shape has been generated, we can evaluate the distribution of the source images over the target. The size, shape and orientation of the source images vary across the target, creating overlaps and gaps in the distribution. Ideally, we would like to obtain a continuous, smooth distribution. We need to avoid under-filling or over-filling the target. In other words, the sum of the areas of the source images should be comparable to the area of the target. We can adjust the number of target points and their location in order to ensure coverage of the target. We developed algorithms that can optimize the tiling of the source images. In Figure 5.19 we compare a target distribution before and after tiling. We only varied the $x$ and $y$ coordinates of the target points. The $z$ coordinate of the target points can also be varied to provide a defocus parameter. This defocus is useful when the size of the source images needs to be increased, which can also soften the edges of the source image.

![Figure 5.19. Optimization of the target grid in order to minimize gaps and overlap.](image)

(a) Initial target grid  
(b) Optimized target grid
5.3.4 **Blending the source images with scattering**

At this point, the distributions we obtained in the previous examples are far from uniform. We obtained a collection of source images with gaps or overlaps, creating obvious peaks and dips in the illuminance distribution. We need to blend the source images together to obtain a smooth distribution. Surface scattering can be used for this purpose. Scattering can be tailored by choosing the reflective surface finish, by using microstructures on the surface, or by adding a diffuser at the aperture of the reflector. In Figure 5.20 we show the effect of Gaussian scattering on the illuminance distribution. The intensity distribution generated by a Gaussian scatterer is defined by

\[
I(\theta) = I_0 \exp \left( -\frac{1}{2} \left( \frac{\theta}{\sigma} \right)^2 \right),
\]

where \( I(\theta) \) is the intensity in the \( \theta \) direction relative to the specular direction, \( I_0 \) is the intensity in the specular direction, and \( \sigma \) is the standard deviation of the Gaussian distribution. We only considered uniform surface scattering over the entire reflector surface. A more flexible but more costly approach is to use a different amount of scattering for each ellipsoid patch, so that we blend areas with overlaps or gaps, while preventing other areas from spreading too much light. Indeed, significant scattering also reduces efficiency by spreading light outside the required target. This can be an issue for applications requiring sharp cutoffs. In this case, the tiling of the source images becomes more critical and requires more accuracy. For many applications however, such as general or automotive lighting, spillover light is not necessarily an issue. Quite the contrary; street luminaires often require smeared edges so that illuminance patterns from
adjacent luminaires can blend their illuminated areas smoothly. The optimum amount of scattering given efficiency requirements can be found with optimization.

Figure 5.20. Effect of scattering on an illuminance distribution. Gaussian scattering with various spread angles $\sigma$ is applied uniformly across the reflector surface. 120×120 bins were used and 2.5 million rays were traced.

5.3.5 Case studies

Street illumination

In this section we show how the method we described is applied to the design of a streetlight reflector. In this first example, the source is placed 30 ft from the ground. The reflector collects light emitted by a cylindrical source over a hemisphere and must illuminate a 120×60 ft rectangular area uniformly. A hole corresponding to a 10° cone is left at the center of the reflector for the light source insertion. The problem geometry and the final reflector design are shown in Figure 5.21. We started with a 20×10 uniform target grid.

After optimizing the system with all previously described degrees of freedom, we obtain the target illuminance distribution shown in Figure 5.22. The corresponding cross-sections for $x = 0$ and $y = 0$ are shown on the right. Uniformity is excellent within the defined 120×60 ft target.
area. The source is a Lambertian cylindrical surface source, 18 mm long and with a 2:1 aspect ratio. The final reflector has a maximum diameter equal to 33.5 cm and is 12.8 cm deep (crossing geometry). We use Gaussian scattering off the reflector surface with $\sigma$ equal to 4.5°.

Figure 5.21. Reflector specifications and final crossing reflector design. The source is a Lambertian cylindrical surface source, 18 mm long and with a 2:1 aspect ratio. The final reflector has a maximum diameter of 33.5 cm.

Figure 5.22. Illuminance distribution on the target. The dotted line represents the limits of the 120×60 ft target area for which the design was optimized. 5 million rays were traced and 80×40 bins were used.
**LED luminaire**

In this second example, the source is placed 100 cm from the target. The reflector collects light emitted by a square surface source (LED chip) from 40° up to 90°, and must illuminate a 50×50 cm square area uniformly. The problem geometry and the final reflector design are shown in Figure 5.23. The reflector geometry is non-crossing in this case. We started with a 10×10 uniform target grid. After optimizing the system, we obtain the target illuminance distribution shown in Figure 5.24. The corresponding cross-sections for $x = 0$ and $y = 0$ are shown on the right. We also obtain good uniformity within the target area in this case. The source is a 1×1 mm Lambertian square surface source. The final reflector has a maximum diameter equal to 23 mm and is 16 mm deep. Gaussian scattering off the reflector surface was set to $\sigma = 1^\circ$.

![Figure 5.23. Reflector specifications and final non-crossing reflector design. The source is a 1×1 mm Lambertian square surface source. The final reflector has a maximum diameter equal to 23 mm and is 16 mm deep.](image)

192
Figure 5.24. Illuminance distribution on the target. The dotted line represents the limits of the 50x50 cm target area for which the design was optimized. 5 million rays were traced and 60x60 bins were used.
CHAPTER SIX - SUMMARY AND CONCLUSIONS

Recent advances in solid-state lighting have pushed the need for compact designs where the extent of the source, however small, cannot be neglected. Numerous illumination tasks in street lighting, automotive lighting, medical lighting, displays, and laser beam shaping require target distributions that can only be efficiently achieved with freeform surfaces. Improvements in manufacturing techniques now enable the fabrication of these complex freeform surfaces, but the major bottleneck now remains the design and testing of these surfaces. Current design methods typically rely on trial and error, which can be time-consuming, costly, and often yields sub-optimal solutions. This research has produced a set of design tools for freeform reflector shapes that take into account the extent of the source and can speed up the design process. The three major contributions of this research are:

- The implementation of an efficient algorithm that generates freeform reflector shapes based on the point source approximation.
- The creation of a method to devise source-target maps that fulfill the integrability condition, which can be leveraged to dramatically speed up reflector generation.
- The implementation of two design methods that take into account the effects of extended sources.

We now describe in detail the contributions in each of these areas. Our first step was to implement a shape generation algorithm based on the point source approximation for systems
without rotational or translational symmetry. While extended sources are our ultimate goal, it is necessary to first understand the design of freeform reflectors with point sources as it already presents major technical challenges. Moreover, the point source approximation often provides valuable starting points that can then be adjusted to compensate for the effects of extended sources. We used the method of supporting ellipsoids developed by V. Oliker as the backbone of our generation method. This method provides solutions for discrete targets made of a finite number of target values. We have devised two efficient implementations of the method of supporting ellipsoids based on increment scaling and direct optimization. Combined with a multi-threaded ray-tracing algorithm, our algorithm allows the creation of reflectors made of 100 ellipsoids in less than 20 seconds. We also suggested improvements that will further reduce computation time. Importantly, we tailored the shape generation algorithm according to the need of the optical designer by letting the designer choose between two distinct reflector shapes that produce the same target distribution (crossing and non-crossing) and by providing a simple way to set the size of the reflector.

However, using the method of supporting ellipsoids with a large number of target points can be computationally prohibitive. We are left with a discrete set of points on the target instead of the desired continuous target distribution. The next step was therefore to understand how to create a continuous target distribution from the discrete target distribution generated by the method of supporting ellipsoids. Direct interpolation of the reflector shape is not effective when the number of target points is small and often results in artifacts in the target distribution. We devised an interpolation technique based on source-target maps that allows to quickly create smooth reflector shapes that produce continuous distributions. This method is especially effective when
the target distribution does not have sharp variations, except at its edges. Additionally, we investigated the behavior of source-target maps and evidenced some interesting and valuable properties:

- A source-target map can only be achieved by a continuous surface if it satisfies the integrability condition. The physical meaning of the integrability condition is to prevent twist in the mapping of the rays from the source to the target.

- Maps that do not satisfy the integrability condition can be achieved by introducing step discontinuities in the reflector shape. However, such discontinuities create artifacts in the target distribution and can make the reflector shape more challenging to manufacture accurately.

- Source-target maps that satisfy the integrability condition can be retrieved with the method of supporting ellipsoids from a small number of target points, as long as the target distribution varies smoothly. Smooth reflectors that produce continuous target distributions can then be generated quickly from these integrable source-target maps.

- Source-target maps tend to vary slowly when system parameters change, as long as these changes do not involve a significant twist in the mapping. This means that reflectors with various system parameters can be generated from the same map. Parameters with low sensitivity include the reflector scale, the target size, and the target aspect ratio. Crossing and non-crossing geometries also often have very similar maps.

Both the shape generation algorithm and the source-target mapping analysis assumed a point source, which forms a useful approximation to obtain first-order reflector shapes. However, most
designs require compact optics and the effects of the extent of the source cannot be neglected. The most common effects of extended sources are a smearing of the target distribution at its edges and either a peak or a dip at its center. We devised two design methods that provide control of the target distribution with extended sources, and we tested their effectiveness for various source sizes and target distribution shapes.

The first design method uses a compensation approach and is suited for systems with rotational or translational symmetry. The prescribed target distribution is used as an optimization variable and is adjusted in order to compensate for the measured effects of the source extent. This approach is effective when the maximum size of the source images on the target is less than about one fifth of the target extent. When the maximum size of the source images becomes too large, the one-to-one correspondence between the source and the target is broken and the compensation approach becomes less effective. Importantly, the compensation algorithm should set maximum allowed target values and should include the extent of the prescribed distribution as an optimization variable, which prevents the algorithm from getting stuck in positive feedback loops.

The second design method for extended sources applies to freeform reflectors. The pinhole source images produced by a reflector generated with the method of supporting ellipsoids are tiled in order to minimize gaps and overlaps, and the source images are then blended together using scattering. In practice, scattering can be introduced by choosing the surface finish, by adding microstructures on the reflector surface, or by adding a diffuser at the exit aperture. This approach is effective when no sharp variations are required in the target, since the use of
scattering tends to smear any high frequency pattern in the target distribution. In order to obtain the desired target distribution, the illuminance of the source images can be adjusted using an illuminance-based shape generation algorithm or by iteratively scaling the target values, similarly to the compensation method used for rotationally symmetric systems. This method is effective as long as the footprint of the source images on the target is smaller than about one fifth of the target extent. The exact domain of effectiveness of the approach highly depends on the specifics of the system parameters and source properties.

In conclusion, we have developed a set of tools aimed at simplifying the design of freeform reflectors with extended sources. These methods have the potential to make the design process more efficient and thus more cost-effective. They enable the design of efficient reflector shapes that minimize light pollution and energy waste. While these methods were applied primarily to single reflective surfaces, the same concepts can be extended to individual facets in faceted reflectors, as well as to refractive surfaces. For refractive surfaces however, special care must be taken to handle potential total internal reflection within the refractive material.

As for the future, three major directions of research are clearly open:

- **Improving the speed of the freeform shape generation algorithm based on point sources.** The current implementation of the method of supporting ellipsoids provides significant speed improvements, but we have shown that even faster convergence can be achieved using an optimization approach (algorithm B in Section 3.2.2). Ray tracing efficiency could be improved by only tracing relevant rays during target evaluation. Indeed, as the algorithm converges, variations in the reflector shape become small and
only a small subset of rays close to the edges of the ellipsoid facets becomes relevant. In addition, using a sequence of reflector shapes with an increasing number of target points, in a fashion similar to multigrid approaches, might help decreasing even further the required computation time. Speed improvement should not be regarded as a trifle: a faster algorithm makes it possible to handle more complex target distributions and makes compensation approaches for extended sources viable since they require the shape generation algorithm to be run multiple times. The shape generation algorithm is the backbone of the design process so every improvement will ease subsequent design steps. Finally, adapting the algorithm for boundary conditions other than a cone would help dealing with the constraints often imposed by packaging requirements or aesthetics concerns.

- **Designing freeform reflectors with realistic source models.** Not only do real sources have an extent, but their emission patterns are highly non-uniform as the result of complex interactions between the emitted light and the opto-mechanical structure of the lamp. We have shown that compensation approaches are successful with systems with rotational or translational symmetry. Systems without symmetry can leverage scattering and tile source images, but this approach can lead to undesirable smearing of the target distribution. Alternatively, integrable maps could be used in conjunction with a compensation approach in order to handle realistic source models and target distributions without symmetry. This way the compensation algorithm adjusts the spread of the flux according to the extended source effects and the map integrability is leveraged to avoid slope errors in the reflector shape.
• **Tolerancing and manufacturing of freeform shapes.** Many discrepancies can occur between the modeled system and the manufactured one: variations in the source emission pattern (especially for LEDs), tolerances on the source position, and accuracy of the reflector shape. As a comparison, the tolerancing and manufacturing of rotationally symmetric aspheric surfaces used in imaging systems, despite having been used for more than 30 years, still pose significant problems, and new asphere descriptions are currently being pushed forward to ease the process [87, 88]. It should thus come as no surprise that the tolerancing and manufacturing of freeform shapes for non-imaging systems remain a scarcely investigated area with virtually no standard approach. While non-imaging systems have in general looser tolerances than imaging systems, a rigorous analysis of the impact of manufacturing tolerances is critical to ensure the robustness and cost-effectiveness of the design. Tolerancing a freeform reflector requires an assessment of the impact of shape variations on the target distribution. As of now NURBS are often used to represent freeform shapes, because they are widely supported across CAD and illumination software. Tolerancing then consists in perturbing the shape descriptors (such as the location of the control points) and evaluating their relative impact on the target distribution. This approach gives an overall “feel” of the sensitivity of the system but does not provide a good estimate of the actual likelihood that such changes will occur during a given manufacturing process. New methods or shape descriptions will need to be derived to do so.
The current transition to solid state lighting brings numerous opportunities to leverage the benefits of freeform reflector shapes in a wide range of applications such as street lighting, automotive lighting, medical lighting [89] and signal lamps used for navigation or air traffic control. An efficient freeform shape design tool is critical to achieve the complex target distributions required for these applications while making the design process cost-effective. New standards against light pollution encourage energy savings and usually require more sophisticated designs, so there is a strong need for modern design tools that help designers and keep design costs low. Finally, systems with strong compactness constraints (close to the étendue limit) or systems that require a control of both the spatial and angular distributions on the target may require more than a single surface to fulfill the design specifications. Extending the current design approach to multiple surfaces could be extremely valuable in these situations.
APPENDIX A - COMPUTATION OF PHOTOMETRIC QUANTITIES
In practice, calculating luminance, intensity or illuminance for arbitrary geometries can be a cumbersome task, especially when these quantities are not constant over space or direction. When the luminance of a surface is constant over direction and space, the emission pattern is called Lambertian. Lambertian emitters are a good 1st order approximation of real sources, and they make calculations of photometric quantities much simpler. Below is a list of common calculations arising in photometry, with their corresponding formulae.

Calculating flux from illuminance: \[ F = \iint E \, dA. \] (A.1)

Calculating flux from intensity: \[ F = \iint I \, d\omega. \] (A.2)

Calculating illuminance from luminance: \[ E = \iint L \cos \theta \, d\omega = \iiint L \, d\Omega. \] (A.3)

Calculating intensity from luminance: \[ I = \iint L \cos \theta \, dA. \] (A.4)

When the illuminance \( E \) is constant, calculating flux from illuminance is equivalent to calculating an area. Similarly, calculating flux from intensity for an isotropic source is equivalent to calculating a solid angle. Finally, calculating illuminance and intensity with a Lambertian source is respectively equivalent to calculating a projected solid angle and a projected area. In fact, we shall see that all quantities can be calculated by performing area calculations.

When illuminance is constant, \( E = E_0 \) and the flux can directly be retrieved from the area of the illuminated region since

\[ F = E_0 \iint dA = E_0 A. \] (A.5)
For areas with arbitrary shapes, it can be useful to use Green’s theorem in order to transform the double integral over the illuminated region $D$ into a path integral over the contour $C$ of that region. The area is then given by

$$A = \iint_D dA = \frac{1}{2} \oint_C x dy - y dx.$$  \hspace{1cm} (A.6)

Similarly, with a Lambertian source, $L = L_0$ and calculating intensity in a given direction is equivalent to calculating the projected area $A_p$ of the light beam in that direction since

$$I(\theta) = L_0 \iint \cos \theta dA = L_0 A_p.$$ \hspace{1cm} (A.7)

This relationship is useful to evaluate the performance of reflectors in the far field. Let us consider a Lambertian source that illuminates a specular reflector. Because of the conservation of luminance, luminance at the reflector surface is equal to the luminance of the source. So in order to know the intensity produced by the reflector in a given direction, one must calculate the projected area of the illuminated region of the reflector in that direction. This region is often called *flashed area* because it corresponds to the illuminated area one would see when looking at the reflector from that direction.

Conversely, if we know the intensity of a Lambertian source in a given direction, we can retrieve its projected area (but not its exact shape). The knowledge of the projected area of the source from its intensity can be useful to speed up computations, as we shall see in Section 4. While it is easy to calculate the area of an arbitrary shape using Green’s theorem, calculating the value of a solid angle or projected solid angle of arbitrary shape is more challenging. We use spherical coordinates, since it usually is a convenient coordinate system for radiating sources. In spherical
coordinates, the expression of a differential solid angle \( d\omega \) and projected solid angle \( d\Omega \) are given by

\[
d\omega = \sin \theta \, d\theta \, d\phi, \tag{A.8}
\]
\[
d\Omega = \sin \theta \cos \theta \, d\theta \, d\phi. \tag{A.9}
\]

From Equation A.3 we see that the illuminance produced on an elementary surface \( dA \) by a Lambertian source of luminance \( L = L_0 \) depends on the projected solid angle of the beam illuminating \( dA \) and is given by

\[
E = L_0 \int \int \cos \theta \sin \theta \, d\theta \, d\phi = L_0 \int \int d\Omega = L_0 \Omega. \tag{A.10}
\]

In order to simplify the integration, we can rewrite the projected solid angle \( d\Omega \) (PSA) using a reference sphere of radius \( R \), as show on Figure A.1. We obtain

\[
d\Omega = \frac{(R \sin \theta)(R \cos \theta \, d\theta \, d\phi)}{R^2}. \tag{A.11}
\]

The quantity \( \rho = R \sin \theta \) is the projected radius of the sphere on the plane \( z = 0 \). We then conveniently have \( d\rho = R \cos \theta \, d\theta \), so the expression of the projection solid angle becomes

\[
d\Omega = \frac{\rho \, d\rho \, d\phi}{R^2}. \tag{A.12}
\]

Since the choice of the radius of the reference sphere is arbitrary, we can use \( R = 1 \) and rewrite Equation A.12 as

\[
\Omega = \int \int \rho \, d\rho \, d\phi \quad \text{with} \quad \rho = \sin \theta. \tag{A.13}
\]

The projected solid angle therefore corresponds to the projected area of the intersection between the reference sphere and the beam that illuminates the differential area \( dA \). We already know to integrate arbitrary area shapes using Green’s theorem. So the only thing we need to do in order to evaluate the illuminance at \( dA \) is to find the intersection between the reference sphere and the
illuminating beam, project its contour on the plane \( z = 0 \), and calculate the corresponding projected area [90]. This is useful to evaluate the performance of reflectors in the near field. As in the intensity case, we simply need to find the flashed area of the reflector for a given point and calculate the corresponding projected solid angle to know the illuminance at that point.

This method is also used in the field of radiative heat transfer in order to calculate configuration factors (also known as shape factors), which are equal to the projected angle divided by \( \pi \) [91]. We can further generalize this method of calculation to solid angles by observing that

\[
d\omega = \sin \theta \, d\theta \, d\phi = \frac{1}{2} \sin \frac{\theta}{2} \cos \frac{\theta}{2} \, d\theta \, d\phi.
\]

Similarly to the projected solid angle case, we can use a reference sphere of radius \( R \), define \( \rho = 2R \sin \theta/2 \), and rewrite the expression of the solid angle \( d\omega \) as
\[ d\omega = \frac{1}{2} \frac{(R\sin \theta/2)(R\cos \theta/2 d\theta)}{R^2} = \frac{1}{2} \frac{\rho d\rho d\phi}{R^2}. \]  \hspace{1cm} (A.15)

Analogously, by setting \( R = 1 \) we end up with a simple formula to calculate any arbitrary solid angle \( \omega \) from the area defined by its projected contour on the reference sphere,

\[ \omega = \iint \frac{1}{2} \rho d\rho d\phi \quad \text{with} \quad \rho = 2 \sin \frac{\theta}{2}. \]  \hspace{1cm} (A.16)

Using this formula, we can now calculate the flux emitted by an isotropic source within any arbitrary solid angle from

\[ F = I_0 \iint d\omega = I_0 \omega. \]  \hspace{1cm} (A.17)

Although there are no analytical formulae for most of the geometries we are dealing with, it is worth mentioning the classical results for the solid angle and projected solid angle of a cone with apex angle \( 2\theta \). Integration can readily be performed in spherical coordinates, yielding

Solid angle:

\[ \omega = 2\pi (1 - \cos \theta), \]  \hspace{1cm} (A.18)

Projected solid angle:

\[ \Omega = \frac{\pi}{2} (1 - \cos 2\theta). \]  \hspace{1cm} (A.19)

The calculation technique outlined in this section is applicable for Lambertian sources. It is possible to extend this technique to non-Lambertian sources by splitting the source into multiple regions where luminance is assumed to be constant, and then calculate the solid angle or PSA for each sub-region. The accuracy of the result will depend on the precision of the source sampling. When the complexity of the computation becomes too high, it is easier to rely on Monte-Carlo methods to evaluate photometric quantities as we shall see in the next section.
APPENDIX B - MONTE-CARLO SIMULATION
B.1 Statistical noise and Accuracy

Monte-Carlo methods are statistical methods used to evaluate functions that are difficult or impossible to calculate analytically. They are well suited to the evaluation of photometric quantities, since the generation of a ray from a source and its interaction with surfaces can both be thought as statistical processes. For instance, let’s imagine that we need to evaluate the flux collected on a surface patch by an optical system. Monte-Carlo ray trace consists of 3 steps:

1. Generation of a sample of rays according to the power distribution of the source
2. Propagation of the rays through the optical system
3. Collection of the rays on the surface of interest

In our example, we assume that the total power of the source is 100 lm, and we trace 100 rays. If 50 rays hit the surface patch, then we can estimate that about 50% of the power emitted by the source falls on the surface patch, i.e. 50 lm.

The accuracy of the result depends on the number of rays traced. If we only traced 3 rays in our example, then the only possible values of the collected flux would have been 0 lm, 33.3 lm, 66.6 lm or 100 lm. We must increase the number of rays to reduce the variance of the estimated flux value. The statistical error of the result is proportional to the standard deviation $\sigma$ of the estimated value, which in turn is inversely proportional to the square root of the number of rays $N_{\text{rays}}$ [92], and is given by

$$\sigma_{\text{noise}} \propto \frac{1}{\sqrt{N_{\text{rays}}}}. \quad (B.1)$$
An unavoidable limitation of Monte-Carlo ray trace is diminishing return: in order to halve the statistical error, we need to trace four times more rays, which greatly increases computational time. So far we considered the evaluation of the total flux collected by a surface patch. But in many situations we need to evaluate an illuminance distribution over an extended surface. This is achieved by dividing the surface into a mesh of bins. Each bin collects rays and yields an estimated value of illuminance. As shown on Figure B.1, we count the number of rays hitting each bin and hence estimate the amount of flux per bin. Using the knowledge of the bin area we can then convert flux into illuminance. The same process applies to intensity, although the use of spherical coordinates induces singularities at the poles, which may require special care.

The statistical error of the estimated value for each bin is proportional to the inverse square root of the number of rays per bin. So there is a tradeoff between the number of bins (the resolution of the illuminance map) and the accuracy of the illuminance values. With a low number of bins, we obtain high accuracy but poor resolution. With a high number of bins, we increase the accuracy
and the statistical error altogether. Figure B.2 illustrates this tradeoff. In this example, the surface to be evaluated is illuminated uniformly. 1,000 rays are traced and rays are collected in a mesh with $5 \times 5$ bins. The corresponding error is about 15%, so the plot looks fairly uniform. If we increase the resolution to $11 \times 11$ bins, the error jumps to 30% and we start seeing noticeable statistical noise on the plot. Increasing the number of rays is the only way to maintain the statistical error while increasing the resolution, as shown on the right of Figure B.2, where 10,000 rays were traced. These error estimates assume a uniform distribution of the rays across the bins. When the estimated distribution is not uniform, more sophisticated estimates of the statistical noise can be made. Overall, statistical errors will be higher where the estimated values are the lowest, since fewer rays are collected.

![Figure B.2. Tradeoff between resolution and accuracy. More rays needs to be traced to maintain the statistical error while increasing the resolution of the receiver where illuminance is measured.](image)

Of great importance is the ability to quantify the spatial uniformity of illuminance (or equivalently, the angular uniformity of intensity) since many tasks require uniform illumination over a given area. We use the relative standard deviation (RSD) of the illuminance as a way to quantify spatial uniformity, defined as

\[ \text{RSD} = \frac{\sigma}{\mu} \]
This is by no means an ideal metric since it does not give any account of spatial patterns that may arise in the evaluated distribution. There is no physiological basis to justify the use of the relative standard deviation as a metric, since the sensitivity of our visual system depends heavily on the spatial frequency of content. It is rather used for convenience and simplicity. Besides, no metrics can encompass the wide range of possible criteria to evaluate uniformity. Metrics should be defined on a case-by-case basis according to the specific task being achieved. The relative standard deviation of illuminance can be directly compared to the statistical noise in order to know if the noise limit has been reached. Once the noise limit has been reached, one must increase the number of rays traced in order to quantify the performance of a system more precisely. For a given mesh resolution and number of rays traced, a distribution is said to be uniform when \( RSD \approx \sigma_{\text{noise}} \).

In practice, a rule a thumb is that good uniformity is obtained when the relative standard deviation of illuminance is below 5%. Smooth variations over large areas can tolerate much greater values of relative standard deviations. On the other hand, the eyes can be highly sensitive to color non-uniformities, especially towards the blue end of the color gamut. In some cases values of RSD as low as 1 or 2% may be required.
B.2 Source modeling

The first step in a Monte-Carlo simulation is the random generation of source rays. In order to obtain the best accuracy for a given number of rays, more rays should be generated where the luminance of the source is higher. This is called importance sampling, which means that the statistical distribution of the generated samples (rays) follows the flux distribution of the source [92]. Another possibility is to generate rays uniformly regardless of the source distribution and then apply a weighting factor on each ray, according to the relative power of the ray. Weighting is easier to implement but typically less accurate since many rays can be generated where power is very low instead of where most of the flux is located. On the other hand, importance sampling can be difficult to implement for arbitrary source power distributions. In order to generate random rays, we need to calculate the inverse cumulative flux distribution of the source [93]. The calculations can be done analytically for ideal sources with isotropic or Lambertian emission patterns but not for arbitrary distributions where the inverse function must be devised numerically.

Let’s first consider the generation of rays for an isotropic source. In spherical coordinates, we cannot simply generate rays equally spaced in $\theta$ and $\phi$ in order to simulate an isotropic distribution. This would result in a higher density of rays closer to the poles, as shown on Figure B.3. We need to derive the direction of the rays based on the flux distribution of the source. As we said, it means calculating the inverse cumulative flux distribution of the source. The normalized cumulative flux of an isotropic source (over a full sphere) is given by
Variables are separable, so we can treat $\theta$ and $\phi$ independently. The cumulative flux in our case is the cumulative probability density function of having a ray emitted in a given direction. Calculating the inverse function for each marginal density yields

$$\phi = 2\pi u_1$$

$$\theta = \cos^{-1}(1 - 2u_2),$$

where $u_1$ and $u_2$ are random variables over the interval [0,1]. A uniform sampling of $u_1$ and $u_2$ over the interval $[\cos^{-1}(\theta_2),\cos^{-1}(\theta_1)]$ generates a set of ray directions according to the flux distribution of the source over the range of angles $[\theta_1,\theta_2]$. Equidistant $u_1$ and $u_2$ values can be used instead of random values, but we then obtain a geometric grid as shown on Figure B.3. This type of sampling can lead to undesirable artifacts and thus random sampling is often preferred.

The same reasoning with a Lambertian source yields

$$\phi = 2\pi u_1$$

$$\theta = \cos^{-1}(\sqrt{1-2u_2}).$$

Random sampling of rays, however, often results in highly non-uniform sampling of the parameter space. It can easily be seen on Figure B.3 that rays tend to cluster in some regions, while leaving other regions empty. One possible solution to this problem is stratified sampling: the parameter space is divided into bins and one ray is generated randomly within each bin as shown on Figure B.3. By doing so, we maintain randomization while ensuring a uniform sampling over space. We used this sampling technique in the simulations presented throughout this dissertation. When dealing with real sources, we used ray data files provided by the
manufacturers, so we do not have to worry about generating rays. Ray data files contain sets of rays modeling the emission pattern of the source and can be readily used is illumination software for simulation.

Figure B.3. Various sampling strategies for sampling rays from an isotropic point source.
The classical equation of an ellipsoid in polar coordinates is

$$\rho = \frac{a(1-e^2)}{1-\mathbf{m} \cdot \hat{\mathbf{v}}},$$

where $a$ is the semi-major axis, $e$ is the eccentricity ($0 < e < 1$), $\hat{\mathbf{v}}$ (unit vector) is the direction of the ellipsoid major axis, and $\mathbf{m}$ (unit vector) is the direction of the polar radius of interest. The quantity $\mathbf{m} \cdot \hat{\mathbf{v}}$ thus corresponds to the cosine of the polar angle $\theta$. The eccentricity $e$ can be expressed as

$$e = \frac{c}{a},$$

where $2c$ is the distance between the two foci of the ellipsoid. Therefore, Eq. C.1 can be rewritten as

$$\rho = \frac{a (1 - c^2/a^2)}{1 - c/a \mathbf{m} \cdot \hat{\mathbf{v}}} = \frac{a^2 - c^2}{a - c \mathbf{m} \cdot \hat{\mathbf{v}}};$$

Using Oliker's notation, we define the focal parameter $d$ as

$$d = a (1 - e^2) = a - \frac{c^2}{a},$$
so the polar equation of the ellipsoid becomes

\[ \rho = \frac{d}{1 - e \mathbf{m} \cdot \hat{\mathbf{v}}}. \]  

(C.5)

We can express the semi-major axis \( a \) from the focal parameter \( d \) and parameter \( c \) as

\[ d = a - \frac{c^2}{a} \]

\[ \Rightarrow a^2 - da - c^2 = 0 \]

(C.6)

\[ \Rightarrow a = \frac{1}{2} (d + \sqrt{d^2 + 4c^2}). \]

We only retain the positive solution to the quadratic equation since \( a > 0 \). Noting that \( 2c = |\mathbf{v}| \),

we can finally rewrite the eccentricity \( e \) as a function of the focal parameter \( d \). We obtain

\[ e = \frac{c}{a} = \frac{2c}{d + \sqrt{d^2 + 4c^2}} = \frac{|\mathbf{v}|}{d + \sqrt{d^2 + |\mathbf{v}|^2}} = \sqrt{1 + \frac{d^2}{|\mathbf{v}|^2} - \frac{d}{|\mathbf{v}|}}. \]

(C.7)

Eq. C.5 and Eq. C.7 correspond to the parameterization used by Oliker in [73]. On the other hand, Eq. C.3 uses the semi-major axis \( a \) as a scaling parameter instead of \( d \), which avoids the computation of a square root.
APPENDIX D – DERIVATION OF MAPPING EQUATIONS BASED ON EQUI-FLUX GRIDS
Analytical source-target mapping equations can be derived when the system has symmetry properties that allow separation of the integration variables. Here we consider the case of a uniform rectangular target with an isotropic point source. The reflector collects light within a cone of arbitrary size. Under these assumptions both the source and the target share radial symmetry so a mapping can be derived by comparing their cumulative flux distributions along $\theta$ and $\phi$. Such mappings will most likely not satisfy the integrability condition but they can still be achieved by introducing step discontinuities in the reflector surface, as detailed in Section 4.1.

We first calculate the source and target cumulative flux within concentric apertures as shown in Figure D.1. Cone-shape solid angles originating at the source are mapped onto rectangular apertures on the target.

Fig. D.1. Concentric apertures used for the calculation of the cumulative flux of the source and the target. Comparing the corresponding source and the target cumulative flux distributions yields the $\theta$ dependency of the mapping.
The flux emitted by an isotropic point source within a cone-shaped solid angle is given by

\[ \Phi_{\text{source}}(\theta) = \int_0^{2\pi} \int_{\theta_{\text{min}}}^{\theta} I_0 \, d\omega = \int_0^{\theta} \int_{\theta_{\text{min}}}^{\theta} I_0 \sin \theta' \, d\theta' \, d\phi' \]  

(D.1)

where \( I_0 \) is the intensity of the source and \( \theta_{\text{min}} \) is the minimum collection angle of the reflector.

On the other hand, the flux contained within a rectangular target aperture is given by

\[ \Phi_{\text{target}}(x_A, y_A) = 4E_0 x_A y_A = 4E_0 \eta y_A^2 = 4E_0 \frac{x_A^2}{\eta}, \]  

(D.2)

where \( E_0 \) is the prescribed illuminance of the target, \( x_A \) and \( y_A \) are the half-width and half-height of the rectangular aperture, and \( \eta \) is the target aspect ratio defined as the ratio between the target width and the target height. To avoid normalization issues between the total source flux and the total target flux we use fractional flux, which is defined as the ratio between the flux contained within a given aperture and the total flux. Comparing the fractional flux of the source and the target yields

\[ \frac{\Phi_{\text{source}}(\theta)}{\Phi_{\text{source}}(\theta_{\text{max}})} = \frac{\Phi_{\text{target}}(x_A)}{\Phi_{\text{target}}(x_{\text{max}})} \]

\[ \frac{\cos \theta_{\text{min}} - \cos \theta}{\cos \theta_{\text{min}} - \cos \theta_{\text{max}}} = \frac{x_A^2}{x_{\text{max}}^2}, \]  

(D.3)

where \( \theta_{\text{max}} \) is the maximum collection angle of the reflector, and \( x_{\text{max}} \) and \( y_{\text{max}} \) are the half-height and half-width of the target, as indicated in Figure D.1. We can therefore express the target aperture half-width \( x_A \) as a function of \( \theta \) as

\[ x_A(\theta) = x_{\text{max}} f(\theta), \]  

(D.4)

where
\[ f(\theta) = \frac{\cos \theta_{\text{min}} - \cos \theta}{\cos \theta_{\text{min}} - \cos \theta_{\text{max}}}. \] (D.5)

The same reasoning can be applied to the target aperture half-height \( y_A \) and yields

\[ y_A(\theta) = y_{\text{max}} f(\theta). \] (D.6)

Eq. D.4 and Eq. D.6 give the \( \theta \) dependency of the mapping. We now need to integrate the source and the target distributions along \( \phi \) in order to obtain the \( \phi \) dependency of the mapping, as shown with the green contours in Figure D.2.

Fig. D.2. Calculation of the cumulative flux along \( \phi \). The target is divided into zones where for a given \( \theta \) either the \( x \) or \( y \) coordinate of the mapping is constant.

Starting from \( \phi = 0^\circ \), integration of the source flux along \( \phi \) for a given \( \theta \) yields

\[ \Phi_{\text{source}}(\theta, \phi) = \int_0^\phi \int_{\theta_{\text{min}}}^\theta I_0 \sin \theta \, d\theta' \, d\phi'. \] (D.7)

\[ \Phi_{\text{source}}(\theta, \phi) = I_0 \phi \left( \cos \theta_{\text{min}} - \cos \theta \right). \]
The cumulative source flux is therefore a linear function of the variable $\phi$. In order to decouple $x$ and $y$ it is convenient to divide the target into 5 zones as shown in Figure D.2. For instance, in zone 1 the $x$ coordinate is fixed at $+x_A$ while the $y$ coordinate varies between 0 and $+y_A$. The cumulative target flux in zone 1 is given by

$$\Phi_{\text{target}}(x_A, y) = \frac{1}{2} x_A y, \quad (D.8)$$

which corresponds to the flux contained within the triangular area highlighted in red in Figure D.2. We see that the target flux is also a linear function of the variable $y$. By comparing the fractional flux of the source and the target, we get

$$\frac{\Phi_{\text{source}}(\theta, \phi)}{\Phi_{\text{source}}(\theta_{\text{max}}, \phi_{\text{max}})} = \frac{\Phi_{\text{target}}(x_A, y)}{\Phi_{\text{target}}(x_{\text{max}}, y_{\text{max}})} = \frac{x_A y / 2}{4 x_{\text{max}} y_{\text{max}}} \quad (D.9)$$

By plugging Eq. D5 in the previous equation, we obtain the mapping for the $y$ coordinate

$$y = y_{\text{max}} \frac{\phi}{\pi/4} \sqrt{\frac{\cos \theta_{\text{min}} - \cos \theta}{\cos \theta_{\text{min}} - \cos \theta_{\text{max}}}}, \quad (D.10)$$

which contains both the $\theta$ and $\phi$ dependency of the mapping. The final mapping equations for zone 1 are thus

$$x = x_{\text{max}} f(\theta), \quad (D.11)$$

$$y = y_{\text{max}} f(\theta) g(\phi),$$

where

$$g(\phi) = \frac{\phi}{\pi/4}. \quad (D.12)$$
The same reasoning can be applied to each target zone. In fact, equations can easily be derived by using the fact that in each zone the cumulative target flux corresponds to a triangular area that varies linearly with either $x$ or $y$. Since the cumulative source flux is linear with $\phi$, the final mapping equations must also be linear with $\phi$ and can directly be inferred. For instance, in zone 2 we know that $y$ is fixed at $+y_A$ and that $x$ varies linearly between $+x_A$ and 0 as $\phi$ increases. More specifically, we must have $x = +x_A$ when $\phi = \pi/4$ and $x = 0$ when $\phi = \pi/2$. From these observations we can infer that the mapping equations for zone 2 must be

$$x = x_{\text{max}} f(\theta) (2 - g(\theta)).$$

$$y = y_{\text{max}} f(\theta).$$

(D.13)

If we assume that $\phi$ belongs to the $[0,2\pi]$ interval, the full set of mapping equations is

\[
\begin{align*}
0 \leq \phi &< \frac{\pi}{4} & x &= x_{\text{max}} f(\theta) & y &= y_{\text{max}} f(\theta) g(\phi) \\
\frac{\pi}{4} \leq \phi &< \frac{3\pi}{4} & x &= x_{\text{max}} f(\theta) (2 - g(\phi)) & y &= y_{\text{max}} f(\theta) \\
\frac{3\pi}{4} \leq \phi &< \frac{5\pi}{4} & x &= -x_{\text{max}} f(\theta) & y &= y_{\text{max}} f(\theta) (4 - g(\phi)) \\
\frac{5\pi}{4} \leq \phi &< \frac{7\pi}{4} & x &= x_{\text{max}} f(\theta) (g(\phi) - 6) & y &= -y_{\text{max}} f(\theta) \\
\frac{7\pi}{4} \leq \phi &< 2\pi & x &= x_{\text{max}} f(\theta) & y &= y_{\text{max}} f(\theta) (g(\phi) - 8).
\end{align*}
\]

(D.14)

Similar relationships can be established for a Lambertian source emission pattern. For a Lambertian source, the intensity is given by $I(\theta) = I_0 \cos \theta$ and the flux emitted within a cone-shaped solid angle is thus

\[
\Phi_{\text{source}}(\theta) = \int_0^{2\pi} \int_{\theta_{\text{min}}}^{\theta} I(\theta') d\omega = 2\pi \int_{\theta_{\text{min}}}^{\theta} I_0 \cos \theta' \sin \theta' d\theta' d\phi
\]

\[
\Phi_{\text{source}}(\theta) = 2\pi I_0 \left(\cos \theta_{\text{min}}^2 - \cos \theta^2\right).
\]

(D.15)
We can therefore still use the formulae in Eq. D.14 with Lambertian sources by simply replacing the original expression of $f(\theta)$ by

$$f_{\text{Lamb}} (\theta) = \sqrt{\frac{\cos \theta_{\min}^2 - \cos \theta^2}{\cos \theta_{\min}^2 - \cos \theta_{\max}^2}}.$$  (D.16)


3. W. Hutchinson, *A treatise founded upon philosophical and rational principles: towards establishing fixed rules, for the best form and proportional dimensions in length, breadth and depth of merchant's ships in general; and also the management of them by practical seamanship* (printed by Thomas Billinge, 1791).


34. V. D. Komissarov, "The foundations of calculating specular prismatic fittings," Trudy VEI 43, 6-61 (1941).


