Optical imaging systems with freeform surfaces : optimization algorithms study and freeform surfaces metrology

Thomas Houllier

N°d'ordre NNT : 2021LYSES004

THESE de DOCTORAT DE L'UNIVERSITE DE LYON
opérée au sein de
l'Université Jean Monnet

Ecole Doctorale SIS488
Sciences, Ingénierie, Santé

Spécialité / discipline de doctorat :
Optique, Photonique, Hyperfréquences

Soutenue publiquement le 07/07/2021, par :
Thomas HOULLIER

Systèmes optiques imageurs avec des surfaces freeform : études d'algorithmes d'optimisation, et métrologie des surfaces freeform

---

Devant le jury composé de :

SORTAIS, Yvan	Pr	Institut d'Optique	Président
ROLLAND, Jannick	Pr	University of Rochester	Rapporteur
DRUART, Guillaume	HDR	ONERA	Rapporteur
TETAZ, Nicolas	Ingénieur	Thales Alenia Space	Invité
LEPINE, Thierry	MC HDR	Institut d'Optique	Directeur de thèse
CASSAR, Guillaume	PhD	Sophia Engineering	Co-encadrant de thèse

Optical imaging systems with freeform surfaces

Optimization algorithms study and freeform surfaces metrology

Thesis author: **Thomas HOULLIER**
August 7, 2021

Thesis advisor: **Thierry LÉPINE**

Affiliations and Funding

The present PhD research work was done while the author was affiliated with:

- Sophia Engineering – 5 rue Soutrane, 06560 Valbonne, FRANCE
- Univ-Lyon, Laboratoire Hubert Curien, UMR CNRS 5516 – 18 rue Benoît Lauras, 42000 Saint-Etienne, FRANCE

The research program is a ANRT CIFRE involving:

- Laboratoire Hubert Curien
- Sophia Engineering
- Thales Alenia Space

It was funded by:

- Sophia Engineering
- ANRT
- Laboratoire Hubert Curien
- Thales Alenia Space

Summary

English

This research work treats the subjects of simulation, automatic design and metrology of freeform imaging optical systems. Starting from an example of freeform unobscured space telescope design, we identified two challenges that are implied or amplified by the introduction of freeform optics: firstly the efficiency of computer-assisted optical design, secondly the possibility of testing the shape of freeform optics. Aspects of performance simulation through sequential raytracing are thoroughly revisited and led to studies on search algorithms applied to freeform optical design. A rigorous methodology from the applied mathematics research field is used to compare the efficiency of various search algorithms in producing high performance solutions to conventional or freeform optical design problems. In a second part, the opportunities presented by phase-measuring deflectometry are highlighted on a freeform mirror sample with characteristics that make it hard to measure with classic metrology means such as interferometry. The great variety of measurement types and samples accessible to deflectometry is demonstrated. A quantitative analysis of the shape measurement error on the freeform sample is performed.

French

Cette thèse a pour sujets la simulation, la conception automatique et la métrologie de systèmes optiques imageurs freeform. A partir d'un exemple de projet de conception de télescope spatial non-obstrué freeform, nous identifions deux difficultés induites ou exacerbées par l'introduction de composants freeform: premièrement la capacité à concevoir rapidement par ordinateur les combinaisons optiques et secondement la capacité à contrôler la forme des optiques freeform. Les aspects de simulation de performance par tracé de rayon séquentiel sont revisités en détail et ont permis des études sur les algorithmes de recherche appliqués à la conception optique freeform. Une méthodologie rigoureuse provenant des mathématiques appliquées est utilisée pour comparer la capacité de différents algorithmes de recherche à produire des solutions d'un haut niveau de performance sur des problèmes de conception optique, conventionnels et freeform. Dans un second volet des travaux, l'intérêt de la

déflectométrie à mesure de phase a été mis en valeur sur un échantillon de miroir freeform présentant des caractéristiques le rendant difficilement mesurable pour les méthodes de métrologie classiques comme l'interférométrie. La grande variété des types de mesures et des types d'échantillon accessibles à la déflectométrie est démontrée. Une évaluation quantitative de l'erreur de mesure de forme de l'échantillon freeform par déflectométrie est effectuée.

Acronyms

ACO Ant Colony Optimization
AR Augmented Reality
BRDF Bidirectional Reflectance Distribution Function
CAD Computer-Aided Design
CGH Computer-Generated Holograms
CMM Coordinate-Measuring Machine
CPU Central Processing Unit
CUDA Compute Unified Device Architecture
DLL Dynamic-Link Library
DM Deformable Mirror
DOF Degrees Of Freedom
EM Electro-Magnetic
FOV Field Of View
FTS Fast Tool Servo
GMT Giant Magellan Telescope
GPGPU General Purpose GPU
GPU Graphics Processing Unit
GUI Graphical User Interface
HMD Head-Mounted Display
HUD Head-Up Display
IBF Ion Beam Figuring
IR Infrared
JWST James Webb Space Telescope
MF Merit Function
MRF MagnetoRheological Finishing
MTF Modulation Transfer Function
NAT Nodal Aberration Theory
NURBS Non-Uniform Rational Basis Spline
PMD Phase-Measuring Deflectometry
PSD Power Spectral Density
PSF Point Spread Function
PSO Particle Swarm Optimization
RBF Radial Basis Function
REPL Read-Eval-Print Loop
RMS Root Mean Square

SBCL Steel Bank Common Lisp
SLM Spatial Light Modulator
SMS Simultaneous Multiple Surface
SNR Signal-to-Noise Ratio
SPDT Single Point Diamond Turning
SSS Slow Slide Servo
SVD Singular Value Decomposition
TIR Total Internal Reflection
TIS Total Integrated Scatter
TMA Three-Mirror Anastigmat
UI User Interface
UV Ultraviolet
VR Virtual Reality
WFE Wavefront Error

Contents

Chapter 1

Introduction

Freeform optical surfaces are surfaces without rotational symmetry [1]. They improve the performance of off-axis optical systems and systems with spatially asymmetric features (*eg* a biased field, offset aperture, *etc*). The performance gains for these systems when compared with conventional spherical or conicoid surfaces can be:

- Improved aberration correction allowing a larger aperture or field of view.
- Greater tilts and decenters of surfaces to reduce and ideally avoid obscuration while maintaining acceptable image quality.
- Improved compacity. Also, complying with exotic packaging geometries.
- Mass reduction by decreasing the total number of required optical surfaces.

Freeform optics is a concept that only became possible due to the interdisciplinary, past and ongoing research in:

- Optical systems design and performance simulation.
- Manufacturing and polishing technologies.
- Optical surface metrology.
- Optical systems integration and alignment methodology.

Freeform optics is now more or less established in some industrial sectors such as airborne imaging systems, space optics, Head-Mounted Display (HMD) and Head-Up Display (HUD) products, varifocal prescription glasses. Lighting systems also commonly exhibit freeform characteristics. The ecosystem is mature enough so that optical designers can propose freeform systems, manufacturers polish the surfaces, metrologists measure them, and systems integrators assemble them. However, research is very much still ongoing in an effort to drive down

the cost and push the envelope with more and more difficult shapes, unlocking even more performance gains.

The present doctoral research work deals with topics in freeform optical systems design, particularly raytracing and search algorithms, and freeform surface metrology. We deal with freeform imaging systems, and especially aim at space optics applications with a focus on unobscured reflective telescopes. Our goal is making the automatic design of freeform telescopes easier for optical design practitioners, as well as investigating freeform surface metrology using phase-measuring deflectometry.

Let us introduce the field of *freeform optics* in a general sense, then which technological and research sub-fields are concerned and how they fit together, and finally let us outline the work we carried out specifically inside this wider field.

1.1 Freeform surfaces and freeform systems

What is meant by the term *freeform* and other surrounding terms? What are examples of the application of freeform optical surfaces?

1.1.1 Freeform?

A *freeform optical surface* is a type of optical surface that constitutes the diopters of lenses or mirrors. Its shape is designed with no axis of symmetry. Planes of symmetry can be found however. By that definition, surfaces described for instance by one of the freeform polynomial bases (more in Chapter 2) are freeform, but off-axis parabolae are not since they have an axis of revolution (albeit outside of their clear aperture). By extension, we refer to optical systems that exhibit at least one freeform surface as *freeform optical systems*.

We call *aspheric* the surfaces that are not spherical and do not fall in the *freeform* category either. These comprise, for instance, conicoids (off-axis or not) and aspheric polynomial surfaces[1]. Surfaces with toroidal or biconic shapes fit the definition of *freeform* we have given, but they are not *typically* refered to as such in the literature and are treated as special cases.

1.1.2 What freeform optics is not

We want to warn against a possible misconception. Freeform optics does not replace spherical optics or make it obsolete. There are numerous designs such as, for instance, refractive photographic lenses that are very unlikely to ever benefit from the addition of freeform surfaces where imaging quality is concerned[2]. Freeform optics is useful only when, for some reason, the optical system is off-axis, such as in unobscured reflective space telescopes (to avoid both

[1] In older papers, the term *asphere* designated all optical surfaces that were not spheres, including freeform. Conventions have since changed.

[2] They do benefit from aspheres however

pupil obscuration and chromatism) or has an asymmetric field. Even for these systems that benefit from freeform optics, the cost and complexity should make the optical designer hesitate and sometimes prefer conventional solutions whenever possible. The difference in technological maturity (mainly in metrology in our view) between freeform optics and conventional optics is the root cause of these difficulties. This gap should hopefully narrow in the future.

1.1.3 Main areas and first systems

Often cited as first commercial freeform optical system is the Polaroid SX-70 which, in 1972, featured an off-axis system with two freeform lenses. The design allowed the camera to be folded for transport [2]. Even earlier, in 1959, the Varilux prescription lenses were designed and produced with a freeform diopter allowing the continuous variation of focal length along the lens surface [3]. Alvarez lenses, the first patent of which was sent in 1964, are also a notable early example. Combinations of two or more freeform lenses produce a system with variable focal length when relative transverse translations or rotations of its parts are applied [4].

Freeform surfaces are used in HMD systems [5, 6], typically in freeform prisms for Augmented Reality (AR) or Virtual Reality (VR) applications [7] and in HUD, for aircrafts or automotive applications [8]. Many lighting applications use freeform surfaces, with design methods that increasingly could be used in imaging optics [9].

As far as space telescopes are concerned, freeform optics is about to be generalized [10]. Off-axis Three-Mirror Anastigmat (TMA) telescopes are already popular choices (with conicoid surfaces), for instance on the James Webb Space Telescope (JWST) [11, 12]. When a sufficient degree of confidence in the metrology, which is critical in space-borne systems, will have been reached, freeform optics is likely to become common in space optics as it should improve the performance and compacity of existing unobscured telescope designs. [13] shows an overview of the DESIS (Deutsche Zentrum für Luft- und Raumfahrt (DLR) Earth Sensing Imaging Spectrometer) instrument, which is an imaging spectrometer with a freeform unobscured TMA as the spectrometer stage. DESIS was launched on June 29th 2018 and was installed on board the ISS [14] (images can be found online). Instances of using freeform optics in space satellites are the TROPOMI telescope [15] (TNO, Netherlands) and the MicroCarb instrument [16]. There are still few examples of spaceborne freeform optics.

1.2 Historical Perspective

> This work is not alone that of an inventor, nor of a telescope-builder, nor of an optician, a glass-maker, a mechanician, a mechanical engineer, a photographic-emulsion maker, an astronomical photographer, a measurer of astronomical photographs, an astronomer. It is not higher than, but *more than* the work of any one of these. It is the work of all of these together.
>
> George Willis Ritchey (1928) [17]

A little perspective on the history of telescope systems can help see how optics researchers were driven to design systems with ever more complex surface shapes. We also see how, each step of the way, progress in every field was necessary: optical design, manufacturing and metrology.

1.2.1 Galileo's era: research in optical instruments

Galileo's *telescopio*, arguably the first notable telescope, the first prototypes of which were produced in 1609, kicks off telescope production. These instruments had a very small aperture (see the dimensions of one of the surviving telescopes [18], the objective lens had a $F\# = \frac{980}{15} = 65$) and reportedly produced very blurry and chromatic results[3].

In the context surrounding the production of these telescopes we already see the need to research all fields in parallel: optical theory as well as manufacturing, engineering and performance modeling. Galileo had read Kepler's treatise *Astronomiae pars optica* [The Optical Part of Astronomy][4] but did not understand key parts of it and proceeded to invent his telescope by labor-intensive trial and error, guided by only rudimentary optical theory (compared to Kepler).This highlights the concurrent nature of manufacturing and theoretical optical design work: each field feeds on the other and sometimes theory does not come before practice.

Additionally we may see in Galileo's story that the applications for his optical instruments are not so different from what they are for cutting-edge optical instruments today. He was funded by the military, sold his telescopes to travelling and sea-faring merchants and carried out astronomical observations[5].

[3] We use the synthesis on the context of Galileo's work by Feyerabend [19] in this section.

[4] It is noteworthy that Kepler introduced in *Astronomiae pars optica* (1604) the unified representation of conics as optical surfaces divorced from the concept of generating cone and intersecting plane[20]. Even in working out his theory, Kepler appears to have thought he was cutting corners for the sake of producing usable concepts for optics, he said: "Atque id est, quod quaerebamus. Caeterùm quia difficilis est consideratio sectionum, proptera quò parum teritur, libet aliqua mecanicè analogicè et populariter de iis differere: date veniam Geometriae [And that is what we were seeking. However, because the subject consideration of [conic] sections is difficult, having been too little pursued, it is permitted to treat them somewhat mechanically, analogically and popularly. Geometers, be indulgent]." (taken from Centina [20]). This corresponds with almost no modification to the parametrized conics representation we still use today in optical design.

[5] The astronomical applications for the telescope left the other academics rather dubitative

1.2.2 Drivers for surface complexity: genealogy of reflective telescope systems

Freeform optics can be seen as the prolongation of the tendancy in optics to use more and more complex surfaces to improve performance, for instance in telescope systems. The drivers for surface complexity were (allowing for great oversimplification) first aberration correction in the center of the field (especially chromatism at first), then increased Field Of View (FOV) and aperture, and finally in the space age, compacity (including non-obscuration).

The first notable reflective telescope is attributed to Newton (1668). The motivation for it was avoiding the chromatic aberrations that plagued refractive designs [21][6]. Indeed, due to a lack of research in the behaviour of the refractive index for available glasses, it wasn't clear at the time whether refractive designs could ever hope to achieve achromatism. Reflective theoretical designs were quite common at the time, even with conics. The designs of Mersenne, Gregory and Cassegrain were produced before or around the time Newton fabricated his first all-spherical reflective telescope. The manufacturing of mirrors was however too far behind the design theory in this case, only concave spherical mirrors could be manufactured with any hope of achieving usable instruments (Cassegrain's convex hyperbola was out of the question) and even then the polished materials (*eg* bronze) produced only dim reflections. It was not until decades after Newton's first prototype that reflective telescopes reached sufficient maturity for them to be used in science observations.

Nonetheless, with the issue of chromatism removed (and spherical aberration solved with conics), larger and larger aperture reflectors were built right up to the beginning of the 20th century thanks, in part, to more mature engineering. Among other factors, we may cite that bronze substrates were abandoned around 1856 and replaced by glass, with better structural stability [22]. In surface metrology, the Foucault knife-edge test (1858) allowed testing optics to within wavelength-range accuracy with great practical ease of use. With the introduction of photographic observation it became necessary to widen the usable field. Ritchey-Chretien telescope designs, solving coma in addition to spherical aberration thanks to their two hyperbolic mirrors, including a difficult convex secondary, allowed such wider fields. Ritchey noted the advantageous compacity of the design, that allowed the whole telescope to fit within a smaller dome [17]. We can also note the concentric Schwarzchild designs, which were an arguably better combination from an aberration point of view[7], but suffered from practical issues in their use (notably the convex field and its overall length).

at first. Indeed Galileo had so far no way of discriminating convincingly between features of his instrument and features of the stars and planets observed. For instance stars were seen with significant chromatic trails due to the instrument's aberrations, which starkly contradicted what even the naked eye could observe. So again we see the parallel nature of the development of this telescope: observations were carried out even before seemingly essential parts of the optical theory were mastered.

[6] We use the work of Baranne and Launay[21] for all the historical facts in this paragraph

[7] "the best combination of two mirrors ever devised, so far as the smallness of the out-of-axis images, alone, is concerned" – Ritchey [17].

Further addition of mirrors allows additional aberration correction, such as astigmatism. The TMA proposed by Paul Maurice (1935) and later improved by Baker (1969) [23] eliminate astigmatism while maintaining also low field curvature. These first theoretical TMA designs were originally on-axis two-mirror corrector solutions with conicoid surfaces for large parabolic primary mirrors. Baker noted the value of this design because it allowed a large field of view in a completely reflective solution. He also noted the potential for space-borne instruments. Korsch showed how to eliminate spherical aberration, coma, astigmatism and field curvature [24] with a TMA solution that can be folded for compacity at the cost of both obscuration and image vignetting [25] (at least in the first such designs, later designs avoid image vignetting). The compacity and straylight performance of this instrument made it popular for space-borne applications.

However, there remains the problem of aperture obscuration, which reduces the quantity of light collected by instruments and degrades the Point Spread Function (PSF), in ways that can be difficult to model. Quite soon in the history of reflective telescopes, Herschel tilted the mirrors of the Forty-Foot telescope (1789) [26] for light collection purposes, as metallic mirrors were not reflective enough. Non-obscured two-mirror telescopes are generally grouped under the Schiefspiegler family, which started, in practice, with Kutter [27].

1.2.3 Compact unobscured reflective telescopes

In contemporary times, for space-borne reflective imagers, one often seeks to avoid obscuration while maintaining high compacity. Since folding an existing on-axis system with flat mirrors necessarily adds either (or all) obscuration, wavefront error, surface diffusion, straylight hazard and weight and causes the need for additional alignment, designers often want to do without. The only two remaining options, as reminded by Fuerschbach in his introduction [28], are then to bias the field and/or offset the aperture (using only part of the field or aperture in an on-axis system and cutting away the useless parts of the mirrors) or alternatively tilt the mirrors themselves starting from an on-axis system. Offsetting the field of an existing rotationally symmetric system allows using only a part of the field away from the center of the aberration field. Offsetting the aperture allows moving the better corrected area of the aberration field into the field that is actually used. Both of these offsets techniques can give rise to very performant unobscured all-conical TMA designs, such as the ones in the JWST [12]. This can be understood using Nodal Aberration Theory (NAT) (more in Section 1.3.1). Tilting mirrors directly allows designing off-axis telescopes divorced from an on-axis parent system, which gives more freedom in terms of system geometry. An early example of unobscured TMA is due to Cook [29], building on the work of Korsch.

Optical surfaces in a tilted system, however, effectively generate aberration fields on the image plane that are not all superimposed around a common center, as is the case for on-axis systems (including systems with offset aperture/biased field). Freeform optical surfaces allow the minimisation of aberrations across the

image field with a higher degree of control over which aberration in particular is being minimized. This in turn enables balancing the non-centered aberration fields generated by each optical surface so that the sum of their contribution on the image plane is minimal on the part of the aberration field that is actually used by the instrument. This is the main motivation for the use of freeform optical surfaces.

1.3 Problems being solved across fields of research

Freeform optics, in a sense, makes every step of optical systems development harder. What are the challenges that freeform optics create and how are they being answered? We generally find these challenges already existed and freeform optics only makes it more critical to solve them. Let us give a brief overview of the state of aberration theory (nodal in particular). The present work does not include the use of nodal aberration theory, but we want to give the reader sufficient information on the existing literature. Secondly, let us give some outline on the more naive approach to optical design via real raytracing and the issues caused by the introduction of freeform optics. This subject occupied most of our research work. Then we will give some information on freeform fabrication, which we left outside the scope of our work. Finally we talk about metrology, which is the object of a few chapters in this memoir.

1.3.1 Aberration Theory

The history of modern aberration theory was summarized by Sasian [30], see also Rolland [31]. The analytical theory suited to model the optical aberrations in tilted, and afterwards in freeform systems, evolved from Hopkins' wave decomposition formula [32], generalized and further applied by Shack. Following the summaries on the subject by Thompson [33] and Fuerschbach [28], the wavefront W in an imaging optical system, depending on the normalized positions in the field $\boldsymbol{H}$[8] and in the pupil $\boldsymbol{\rho}$ can be decomposed into an addition of surface contributions $W_j(\boldsymbol{H}, \boldsymbol{\rho})$ (Eq. 1.1).

$$W(\boldsymbol{H}, \boldsymbol{\rho}) = \sum_j W_j(\boldsymbol{H}, \boldsymbol{\rho}) \tag{1.1}$$

These contributions can be rewritten as the sum of an expansion on the dot products of the position vectors (Eq. 1.2), see eq 2 in Thompson [33].

$$W_j(\boldsymbol{H}, \boldsymbol{\rho}) = \sum_p \sum_n \sum_m W_{klm,j} (\boldsymbol{H} \cdot \boldsymbol{H})^p (\boldsymbol{\rho} \cdot \boldsymbol{\rho})^n (\boldsymbol{H} \cdot \boldsymbol{\rho})^m \tag{1.2}$$

With $k = 2p + m$ and $l = 2n + m$. The scalar W_{klm} is a wave aberration coefficient indexed by k,l and m. These indices correspond to the field, aperture

[8] We use a bold notation for vectors.

and radial power dependency of each aberration's wavefront. The less general (reducing the field to a single dimension for working in rotationally symmetric systems) scalar expression (eq 1 in [33]) highlights these dependencies best (Eq. 1.3).

$$W_j = \sum_p \sum_n \sum_m W_{klm,j} H^k \rho^l \cos^m(\phi) \tag{1.3}$$

With $H = \|\boldsymbol{H}\|$, $\rho = \|\boldsymbol{\rho}\|$ and ϕ the angular position in the aperture. Thus, the familiar power term, which is a parabolic wavefront, is represented by $W_{020}\rho^2$, coma by $W_{131}H\rho^3\cos(\phi)$ (which we know is linear in the field and cubed in aperture position), and so on and so forth.

The above representations are suited to on-axis spherical systems, Buchroeder [34] saw that each optical surface contributed its own aberration field to the final wavefront, and that these fields were off-centered depending on the misalignments in the system. The first NAT paper according to Thompson is [35], where the misalignment of two-mirrors telescope (Cassegrain, Ritchey-Chrétien *etc*) is studied and shown to generate *binodal astigmatism* for the wavefront in the exit pupil. The theory was then expanded to account for freeform surfaces in off-axis systems and fifth order terms were derived [33, 36, 37]. A useful summary for NAT is provided by Thompson [38]. It highlights that the central idea to NAT is the introduction of a vector multiplication operation from Geometric Algebra. This allows rewriting the aberration field expressions in ways that clearly highlight nodal behaviour of aberration fields as a function of the tilt of the components in a system.

We also want to cite Sasian [30] as a reference in the study of optical aberrations. Readers can consult [39, 40] for examples of the application of NAT to optical design tasks.

1.3.2 Optical design

NAT is a valuable tool for optical design, as it provides an *analytical* model for the image performance of a system across the field. This allows compensating aberrations directly, instead of applying an iterative search as is common in optical design. Even in the case of the application of search algorithms (more on that later), an analytical model linking performance with variables such as component tilts, radii of curvature *etc*, is valuable since it gives access to derivatives to guide the search.

1.3.2.1 Real raytracing

Optical systems design, in most cases, is done with constraints in addition to image quality. The designer seeks the best compromise between these constraints and image quality. General search algorithms are thus required. Another general tool is so-called *real* raytracing, which models the imaging performance of a system via the tracing of rays obeying the Snell-Descartes laws of refraction/reflection (for simple systems at least). The real raytracing approach is

largely independent from aberration theory. It is more general although it gives less information to the optical designer. For instance, real raytracing can be used for wavefront simulation down to the nanometer range or to estimate the performance of optical systems the surfaces of which were measured and are represented by grids of points with a high spatial frequency. It is also simple conceptually (though complex to perform as we will see). Real raytracing is the favored approach in most optical design software ever since the field took off in the late 20th century with the advent of powerful computers[9].

Although the study of aberration and real raytracing are different approaches, an expert designer is likely to use either or both depending on the project he is working on.

1.3.2.2 Freeform and search algorithms

The typical workflow of an optical designer assisted by a software is: set up an accurate quantification of project requirements that are obtainable via simulation (such as real raytracing), define *variables* in the optical system definition (such as radii of curvature, tilts, thicknesses) the value of which will be modified in order to meet the requirements, finally run search algorithms on the search problem that was set up. The software will output a new set of variables that generates an optical system that is closer to satisfying the project requirements inputted by the designer. The designer will then tweak the inputs, perhaps manually correct some aspects of the design that cannot be simulated or quantified easily for the search algorithms to process, and run the search again. The designer will iterate these steps until he is satisfied with the optical system, both regarding the easily quantifiable requirements and the requirements that can only be (easily) assessed by a human.

With the increase in computing power, one of the winning strategies for optical designers has been to craft more and more intricate and comprehensive quantifications of the project requirements, in order to automate more and more of the design process. Likewise, the designer wants the software to operate on a maximum number of variables at once, to find the best possible result. This strategy works as long as the dimensionality of the search space (the number of variables) is relatively low (say a dozen in the current state of software).

The introduction of freeform optical surfaces in this workflow increases dramatically the number of variables the search algorithms need to operate on. A single surface can be defined with dozens of coefficients, bringing the total number of variables in an optical system in the hundreds. Search algorithms designed for lower dimensionalities will behave poorly in higher dimensions. In addition, the performance simulation itself is made harder to perform, both conceptually (the software itself is harder to program) and computationally (the computations are costlier for the end user). The result is that the existing strategy outlined above is severely impeded.

[9] Paraxial rays were traced in the early days of the discipline. The term *real* for raytracing refers to the fact that rays obey Snell-Descartes laws instead, which model more accurately the real behaviour of light.

There are several axes of improvement:

- Keep upgrading the computation power. The issue is that these improvements are largely linear while our problem complexities likely grow in superlinear fashion[10].
- Search algorithms better suited to high dimensionality. They would require a lesser computational cost to find solutions equally as good when compared with existing search algorithms (or better with the same invested computational cost).
- Draw more information from the simulation, for instance with aberration theory.
- Obtain accurate performance estimates at a lesser cost. This is for instance the role of aperture stop sampling schemes, which we will detail in this work and that are currently used in optical design.

1.3.3 Fabrication

The manufacturing of freeform surfaces is enabled by a variety of technologies, with varying cost/quality compromises and degree of generality in the generated surfaces. High level reviews such as [1, 41, 42] synthesize this information.

- *Single Point Diamond Turning (SPDT)* is able to produce freeform optical surfaces from even very rough starting points. The final quality of the optical surfaces, both in roughness and shape is increasingly good and even compatible with imaging for some applications, either in Infrared (IR) [43] or with low image quality requirements (See freeform prism [44]) and more recently for space optics [45]. The machines mainly use Slow Slide Servo (SSS) (also called Slow Servo Tool) [46] with a tool axis (Z) perpendicular to the sample as well as an axis (X) in the spindle plane that both oscillate synchronously with the spindle rotation to produce freeform optics. Another method is Fast Tool Servo (FTS) [47] where the tool itself (and not the whole tool table) is driven by much lighter and higher frequency drivers such as piezo actuators or voice coils.
- *Molding* is adequate for low-cost applications and plastic optics, such as in VR and HMD [48] or lighting.
- *Robotic Polishing*: the polishing tool is borne by a robotic arm.
- *Stressed Mirror Polishing* [49]: The substrate is mechanically stressed with a specific deformation. While this constraint is applied, the surface is polished using traditional means into a shape that is easy to manufacture

[10] Say hypothetically that the cost of finding a good solution to a search problem is k^n with k an arbitrary cost and n the number of search variables. Throwing twice as much computing power (a massive improvement in computer hardware) at the problem does not give access to problems with much higher values of n.

(such as a sphere). Once the polishing is done, the release of the mechanical constraints creates the final optical surface shape.

- *MagnetoRheological Finishing (MRF)* [50] uses an abrasive fluid, the viscosity of which can be modified locally via a magnetic field.
- *Ion Beam Figuring (IBF)* [51, 52] projects an ion beam onto the surface to polish. The ions produce material ejection locally on the substrate.
- *Additive Manufacturing* [53–57] is a field covering many technologies. There is a wide variety of applications, from micro-optics to space mirrors. The materials can be transparent plastic or more rarely glass. Mirrors can be made out of metallic (aluminum, titanium) or ceramic substrates. The main advantage is being able to print the optical surface in the same material and in the same manufacturing step as the mechanical structure behind it. Generally speaking, the maturity of these technologies and processes is lower than for the other manufacturing technologies listed here.

The interesting criteria to select which method to use are the cost (equipment required, man-hours required), time (driven in part by the removal rate of the method), final surface roughness, shape accuracy, shape limits (maximum manufacturable slope or altitude departures, surface diameter), the range of materials available.

1.3.4 Surface Metrology

Freeform optical surface metrology is a critical step in freeform optics. Depending on the amplitude of the surface departures, the base shape, the surface shape error budget, the piece diameter, the measurement cost/time, one must choose the adequate metrology method. We have reviewed the working principles and comparative advantages of metrology methods adequate for freeform optics in Chapter 8. See [1] for a review.

Particularly in metrology, surfaces were always "freeform", although this was an unwanted artefact of manufacturing up to a recent date. The challenge we face is that freeform departures are more and more pronounced as optical designers become more acquainted as a profession with freeform optics, and as the number of manufacturers equipped with freeform-ready machines and expertise grows. This is an issue since most conventional metrology methods, such as Fizeau interferometry (which is perhaps one of the most trusted methods for conventional optical surface metrology), rely on the comparison with spherical calibers. The measurement range of most conventional metrology methods is not adequate for freeform metrology. This is why, in our work, we investigated deflectometry as a promising method for the measurement of even strong freeform surfaces.

1.3.5 System Alignment

Optical system alignment is another hard discipline made arguably harder by the introduction of freeform optics. In the case of telescope systems, the goal is to decenter and tilt the mirrors in order to obtain minimized aberrations across the field (which should be close to the nominal design performance if we ignore the surface shape errors). Since we solve an inverse problem not unlike that of optical design, we can use conceptually the same methods. However, since the acquisition of the image aberration field takes time (typically with an interferogram at each point in the field or with a star field), as does translating and rotating the mirrors, black-box optimization approaches are heavily penalized and NAT becomes indicated. NAT allows linking analytically the mechanical decenters and tilts of each mirror to separable effects on the aberration field. This allows a guided alignment procedure: see Schmid [58, 59] for two mirrors and Thompson [60] for TMA alignment. See Hampson [61] for an example of a more empirical automatic alignement system for two-mirror telescopes. See also the proposed methods for aligning the JWST segments using a star field [62]. Using well thought-out mechanical fiducials for obtaining the best possible alignment from the assembly alone before starting the fine alignment procedure helps of course [28].

1.4 Thesis Summary

The present document is divided into chapters as follows:

1. We first review the types of freeform surfaces representations and brush many related topics.

2. An overview of freeform unobscured telescopes is given. We include both a review of systems from the literature and typical concepts that prove useful in designing this kind of systems.

3. In application of the design concepts for freeform unobscured telescopes, we design a freeform TMA for thermal IR Earth observation. We give the prescription data and all relevant performance results. Because designing freeform systems is hard and time-consuming with conventional tools (and without the help of aberration theory methods), we decided to dig deeper into raytracing and search algorithms in the following chapters.

4. We remind the reader of all the applied mathematics involved in sequential raytracing, or at least enough to be able to model most components and systems, including freeform unobscured telescopes. We implemented all of these concepts in our own sequential raytracer.

5. We dig even further into optical design software by explaining many concepts relating to ray-aiming. We give some results and many ideas for the implementation of this hard task made even harder by freeform optics.

6. Now armed with our raytracing engine and no lack of software optimization work, we run computationally-intensive experiments on search algorithms from the applied mathematics literature. We compare the results of minimization runs on conventional and freeform systems between many search algorithms and a commercial optical design integrated solution. Additionally, we give a clear account of search algorithms concepts to help optical designers understand this crucial part of the tools they use. The raytracing implementation plus search algorithms constitute the outline of a simple self-standing optical design program.

7. In a second, smaller part of our work, we have worked on the metrology of freeform optical surfaces. We start by giving a review of the measurement principles which are, in our view, the most able to output low to mid spatial frequencies shape maps of freeform surfaces.

8. We give the results of measurements we carried out using a phase-measuring deflectometry bench on various samples. The prominent results are those on a freeform torture test mirror we designed. This mirror exhibits extreme altitude and slopes freeform departure.

Chapter 2

Freeform Representations

Mathematical freeform representations – 2D polynomials – Point-based – RBF and local representations – Orthogonality properties – Fitting – Generalized evaluation via Clenshaw's method – Edge cases – Suitability v. application

We provide a high level review of the different ways to wholly describe freeform optical surfaces continuous in altitude and slope. We refer to similar reviews by Gross *et al* [63], Ye *et al* [64] and Steinkopf *et al* [65][1].

We distinguish three broad families of representations:

- 2D polynomials
- Point-based
- Radial basis functions and local series expansions.

We describe these representations as well as provide some information concerning computerized implementation. We emphasize polynomial representations since they are typically more useful when used in conjunction with search algorithms for the automatic design of optical systems. This emphasis, or bias, is also found in the discipline as a whole. We discuss some common operations such as polynomial fitting, normalization and shape pre-factors. We give some important mathematical generalizations for freeform polynomials: Clenshaw's evaluation method as well as a general outlook on how new polynomials could be generated for optics. We conclude with some synthetic and practical remarks concerning freeform representations.

2.1 2D Polynomials

Freeform optical surfaces can be described using analytical formulae with 2D polynomials. We refer here directly to the work of Brömel *et al* [66, 67] which

[1] It should be noted that this kind of review starts becoming less and less necessary as their number grows in the literature. Nonetheless, for the sake of completeness and because there were many related subject we wished to include, here is our own review.

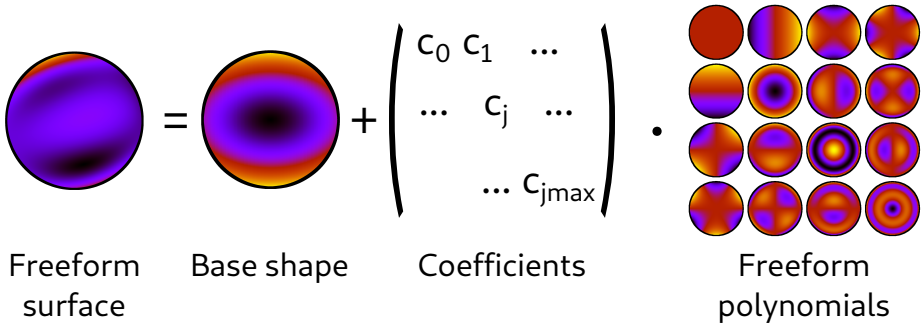


Figure 2.1: Illustration of the decomposition principle of polynomial-based freeform representations. The freeform shape is the result of the addition of a base shape and a weighted sum of polynomials. The dot symbol denotes the scalar product between the coefficients and the polynomials.

compiles a summary of these representations. A common characteristic of all polynomial representations is to be defined using a base shape to which is added a weighted sum of 2D polynomials, as illustrated in Fig. 2.1. The altitude $z(x, y)$[2] along a surface defined with a polynomial-based representation is expressed in Eq. 2.1, with P_j arbitrary 2D polynomial terms, and c_j coefficients weighting their sum.

$$z(x, y) = z_{\text{base}}(x, y) + z_{\text{poly}}(x, y) \tag{2.1}$$

$$z_{\text{poly}}(x, y) = \sum_{j=0}^{jmax} c_j P_j(x, y) \tag{2.1a}$$

2.1.1 Properties in Common

Polynomial-based representation share a number of characteristics which we summarize here.

2.1.1.1 Base Shapes

The base shape component of the description is used in optical design or manufacturing as a link to conventional optical surfaces. The optical designer will often have a starting point that uses conventional shapes. The base shape component allows the representation of these shapes. It can also be used to straightforwardly quantify and limit the freeform departure from a base shape that would be easier to manufacture. The commonly used base shapes are: planar, spherical, conic, toroidal, biconic (Eq. 2.2). All these definitions can be found in [67] and in the user manual of any optical design software [68].

[2] Or $z(\rho, \theta)$. Please note that throughout the present chapter, we will use cartesian or polar coordinates interchangeably.

$$z_{\text{base}}(x,y) = \begin{cases} 0 & \textit{planar} \\ \frac{c\rho^2}{1+\sqrt{1-c^2\rho^2}} & \textit{spherical} \\ \frac{c\rho^2}{1+\sqrt{1-(1+K)c^2\rho^2}} & \textit{conic} \\ \frac{c_x x^2+c_y y^2}{1+\sqrt{1-c_x^2x^2-c_y^2y^2}} & \textit{toroidal} \\ \frac{c_x x^2+c_y y^2}{1+\sqrt{1-(1+K_x)c_x^2x^2-(1+K_y)c_y^2y^2}} & \textit{biconic} \end{cases} \tag{2.2}$$

With:

- $\rho = \sqrt{x^2+y^2}$
- $c = \frac{1}{R}$: the curvature of the sphere of radius R.
- c_x, c_y: the curvatures in X and Y directions.
- K: the conic constant.
- K_x, K_y: the conic constants in X and Y directions.

The biconic base shape is the most general description. However, discernment should be exercised when choosing a description depending on the context. Introducing superfluous Degrees Of Freedom (DOF) is detrimental to optical design optimization. An unnecessarily complex base shape can be a source of confusion when trying to communicate the surface data. One can also introduce degeneracies in the surface description (see Takaki *et al* [69]), meaning that the base shape and some freeform terms compensate each other. An obvious example is the spherical shape both in base shape and Zernike polynomial $Z_{n=2}^{m=0}$ but degeneracies also appear with conics.

2.1.1.2 Multiplying Polynomials with Shape-Controlling Factors

As explained by Brömel [66] (we refer the reader to her memoir for further details and only reproduce a quick summary here), one can introduce 2D shape-controlling factors to the freeform description: namely a *boundary function* $B(x,y)$ and a *projection factor* $P(x,y)$[3]. Note these spatial-dependent factors are simply a way of manually constraining freeform departures on a case by case basis, they are not to be confused with the *weight function* (see Section 2.1.1.5) that characterizes the orthogonality of polynomials. The general expression for polynomial-based freeform surfaces is then Eq. 2.3.

$$z(x,y) = z_{\text{base}}(x,y) + B(x,y)P(x,y)z_{\text{poly}}(x,y) \tag{2.3}$$

[3]Please note that our projection factor is the reciprocal of the projection factor defined by Brömel [66]. This change is for readability. Additionally, it seems to us there are small confusions in the equations between the projection factor and its reciprocal in [66], eq.2.8 for example.

The boundary function $B(x, y)$ provides an envelope for the freeform shape. One can thus constrain, for example, the border of the unit disk to have zero freeform departure, to avoid the often extreme and sometimes useless departures that many polynomials present there. Brömel provides an example of such a boundary function, constraining the center and edges of the unit disk to be zero, we propose a slight variation with a factor of 4 to maintain a unit maximum over the unit disk in Eq. 2.4. It is plotted in Fig. 2.2.

$$B_{\text{center}-\text{edge}}(\rho) = 4(1-\rho^2)\rho^2 \tag{2.4}$$

We can also add piecewise boundary functions with smooth transitions which can be used to spatially knit together arbitrary freeform representations on different areas of a surface or simply to smooth out the edges for better manufacturability. The smoothstep functions provide a transition from 0 to 1 guaranteeing zero derivatives at the edges from order 1 through n[4]. This property allows manufacturing transition areas without sudden changes in slope or tool acceleration. Smoothstep functions are an application of Hermite interpolation, the general analytical expression is found in Eq. 2.5 [70], an application for order $n = 2$ (zero slope and curvature at the edges) is found in Eq. 2.6. Wikipedia provides a list of many other *sigmoid* functions, some of which could probably just as well be used [71] for the same purposes.

$$S_n(x) = x^{n+1} \sum_{k=0}^{n} \binom{n+k}{k} \binom{2n+1}{n-k} (-x)^k \tag{2.5}$$

$$S_2(x) = 6x^5 - 15x^4 + 10x^3 \tag{2.6}$$

Using this sigmoid function, we can craft a piecewise edge-smoothing function. It can be useful to knit the outer border of an optical surface with a mechanical flat outside of the rays footprint. For example, we may wish to transition from a freeform to the base shape between $\rho = \rho_{\text{trans}}$ and $\rho = 1$ on the unit disk, we may use $B_{\text{S}_2}^{\rho_{\text{trans}}}(\rho)$ Eq. 2.7. It is plotted in Fig. 2.3.

$$B_{\text{S}_2}^{\rho_{\text{trans}}}(\rho) = \begin{cases} 1 & \rho < \rho_{\text{trans}} \\ S_2(1 - \frac{\rho - \rho_{\text{trans}}}{1 - \rho_{\text{trans}}}) & \rho_{\text{trans}} \leq \rho \leq 1 \end{cases} \tag{2.7}$$

The projection factor $P(x, y)$ has the effect of dampening the freeform departure in relation with the normal vector angle of the base shape. The freeform departure is projected along the normal vector of the base shape $\overrightarrow{N_{\text{base}}}$ rather than along the piece Z axis. The expression for $P(x, y)$ is simply Eq. 2.8 with α the angle between the base shape normal vector and the Z axis.

[4] One should probably avoid using high orders of n due to accuracy concerns, as the computation of the function involves an addition of powers. We found $n = 2$ to be sufficient for practical purposes, although we did not carry out formal investigations

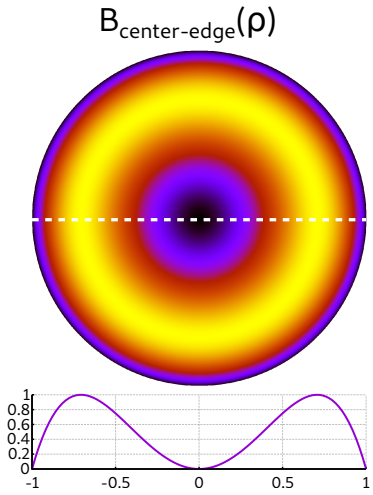


Figure 2.2: Surface plot and diametral cross-section of $B_{\mathrm{center-edge}}(\rho)$ Eq. 2.4. The center and edges of the surface shape are constrained to zero.

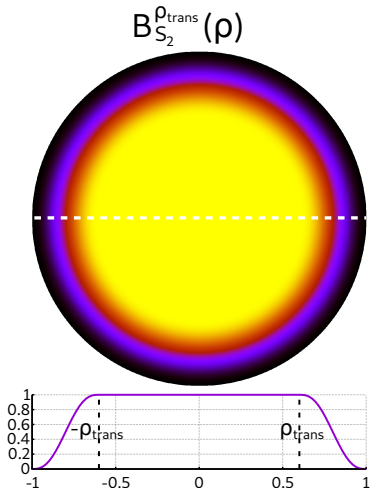


Figure 2.3: Surface plot and diametral cross-section of $B_{\mathrm{S_2}}^{\rho_{\mathrm{trans}}}(\rho)$ Eq. 2.7. The edges after ρ_{trans} transition smoothly to zero.

$$
\begin{aligned}
P(x,y) &= cos(\alpha) \\
&= \frac{\overrightarrow{N_{\mathrm{base}}}}{\left\|\overrightarrow{N_{\mathrm{base}}}\right\|} \cdot \overrightarrow{z} \\
&= \frac{1}{\left\|\overrightarrow{N_{\mathrm{base}}}\right\|} \begin{bmatrix} \frac{\partial z_{\mathrm{base}}(x,y)}{\partial x} \\ \frac{\partial z_{\mathrm{base}}(x,y)}{\partial y} \\ -1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} \\
&= \frac{1}{\sqrt{1 + \left(\frac{\partial z_{\mathrm{base}}(x,y)}{\partial x}\right)^2 + \left(\frac{\partial z_{\mathrm{base}}(x,y)}{\partial y}\right)^2}}
\end{aligned}
\tag{2.8}
$$

In the case of a spherical base shape of curvature c, following Brömel [66] we then have $P_{\mathrm{sphere}}(\rho)$ (Eq. 2.9). We plot this profile in Fig. 2.4.

$$
P_{\mathrm{sphere}}(\rho) = \sqrt{1 - c^2\rho^2} \tag{2.9}
$$

The projection factor has the advantage over the boundary factor to be adaptable on the base shape. In optical raytracing, avoiding important departures near the edge of very curved base shapes will tend to reduce the probability of rays missing the surface, which typically generates an error in the merit function computation in raytracing software.

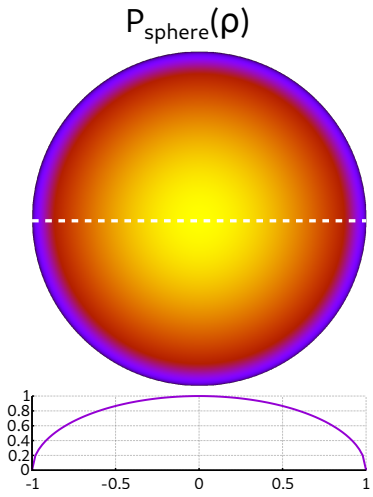


Figure 2.4: Projection factor $P_{\text{sphere}}(\rho)$ (Eq. 2.9) on a whole hemisphere. The freeform shape is contrained to zero as the normal vector approaches $\frac{\pi}{2}$.

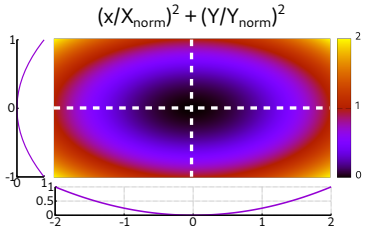


Figure 2.5: Simple example of adapting XY polynomials to a rectangular aperture with normalization values $X_{\text{norm}} = 2, Y_{\text{norm}} = 1$.

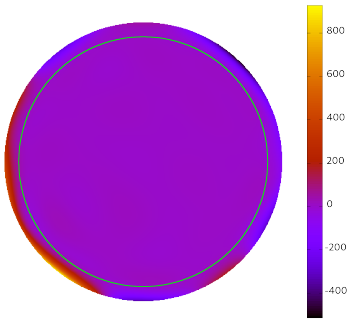


Figure 2.6: Random Zernike polynomials surface evaluated outside of the unit-disk. The unit-disk boundary is indicated by the green line. The outer region has altitude departures which are completely out of proportion with those in the unit-disk.

2.1.1.3 Coordinates Normalization

Polynomials being usually defined either on the unit disk or the unit square, they are generally used with normalized coordinates. One can also want to adapt a polynomial to a different aperture shape, as in Fig. 2.5. In the particular case of XY polynomials, we avoid extreme values outside of the normalized unit square by a simple change of coordinates. Simple coordinates normalizations include:

- Normalization radius (Eq. 2.10), for the normalization of *circular* aperture shapes.
- X-Y normalization (Eq. 2.11), for the normalization of *rectangular* aperture shapes.

$$z_{\text{poly}}(\hat{x}, \hat{y}) = z_{\text{poly}}\left(\frac{x}{R_{\text{norm}}}, \frac{y}{R_{\text{norm}}}\right) \tag{2.10}$$

$$z_{\text{poly}}(\hat{x}, \hat{y}) = z_{\text{poly}}\left(\frac{x}{X_{\text{norm}}}, \frac{y}{Y_{\text{norm}}}\right) \tag{2.11}$$

2.1.1.4 Aperture Shape and Extrapolation

Most polynomials can be extrapolated outside of their support. This is an important property in optical design optimization. The diameter of optics can vary automatically during optimization but the polynomials will continue to be evaluated even if the diameter grows beyond the normalization radius. This prevents unevaluable merit functions.

For "production-ready" surfaces however, extrapolated regions will, as a general rule, have undesirably large departures, as can be seen on Fig. 2.6. It is often recommended to define polynomials over a larger normalization radius than what is needed for the clear aperture, in order to describe the necessary mechanical outer region with polynomials that are not extrapolated. A better way still in our opinion is to transition from the freeform surface to a mechanical flat using a *sigmoid* function as mentioned above (Section 2.1.1.2). The surface will then have to be communicated to the manufacturer in a point-based representation, which can degrade the accuracy if one is not careful.[5]

Adapting polynomials to other aperture shapes, while retaining their orthogonality, can be done via a Gram-Schmidt process [72].

2.1.1.5 Polynomials Orthogonality

Definitions We explain what is meant when polynomials are called *orthogonal* Eq. 2.12 or *slope orthogonal* Eq. 2.13 for any two polynomial terms $P_j, P_{j'}$

[5] In our experience, manufacturers do not like instructions such as "Use the freeform coefficients over the clear aperture and then transition smoothly to a flat on the outer diameter of the piece." It is better in practice to provide them a unique definition for the whole piece.

taken in an single-indexing scheme and for a unit *weight function*. We give the definitions for the unit disk, but they can be applied to any 2D support.

$$\langle P_j, P_{j'} \rangle = \frac{1}{\pi} \int_{\theta=-\pi}^{\pi} \int_{\rho=0}^{1} P_j(\rho,\theta) \cdot P_{j'}(\rho,\theta) \rho d\rho d\theta = \text{const}(j)\delta_{j,j'} \tag{2.12}$$

$$\langle \overrightarrow{\nabla} P_j, \overrightarrow{\nabla} P_{j'} \rangle = \frac{1}{\pi} \int_{\theta=-\pi}^{\pi} \int_{\rho=0}^{1} \overrightarrow{\nabla} P_j(\rho,\theta) \cdot \overrightarrow{\nabla} P_{j'}(\rho,\theta) \rho d\rho d\theta = \text{const}(j)\delta_{j,j'} \tag{2.13}$$

$\delta_{j,j'}$ is the Kronecker delta. If additionally, $\text{const}(j) = 1$, then the polynomials are said to be *orthonormal*.

One can also express orthogonality with an integral scalar product with a non-unit weight function $W(\rho,\theta)$: Eq. 2.14. The resulting mathematical definition of the Root Mean Square (RMS) will then be weighted and will not express the departure in nanometers over the surface to which optics practitioners are accustomed.

$$\langle P_j, P_{j'} \rangle = \int_{\theta=-\pi}^{\pi} \int_{\rho=0}^{1} P_j(\rho,\theta) \cdot P_{j'}(\rho,\theta) \cdot W(\rho,\theta) \rho d\rho d\theta \tag{2.14}$$

The orthogonality property enables, in practice, three very related applications:

- Computing the RMS departure (altitude or slope depending on the type of orthogonality) from the indiviual coefficients alone.
- Easily fitting a surface with polynomials, through a separable minimization problem.
- A very powerful evaluation algorithm (see Section 2.1.3.1).

Formally and in the most general sense, the set of polynomials with degree at most n is a vector space P_n of dimension $n+1$. An inner product $\langle \cdot, \cdot \rangle$ can be defined between terms (it is an integral over some domain in the examples above). The inner product is a scalar function of two members of P_n and has the following properties [73]:

- Symmetry: $\langle P_i, P_j \rangle = \langle P_j, P_i \rangle$
- Linearity in the first argument: $\langle aP_i, P_j \rangle = a\langle P_i, P_j \rangle$ and $\langle P_i + P_k, P_j \rangle = \langle P_i, P_j \rangle + \langle P_k, P_j \rangle$
- Positive-definite for any P_i except 0: $\langle P_i, P_i \rangle > 0$

The integral over a surface of the product of two altitudes is an inner product, the same integral for the dot product of gradients is another inner product. Adding a weight function to the integral still defines an inner product *etc.*

In the most general sense, saying that a polynomial basis is orthogonal is just stating that there exist an inner product that is zero when applied to any two different members of the basis: Eq. 2.15.

$$\forall(i,j) \text{ so that } i \neq j, \langle P_i, P_j \rangle = 0 \tag{2.15}$$

For freeform polynomials we should be careful to specify what inner product is meant when we call polynomials *orthogonal*, since it is often linked to physical quantities we care about.

Polynomials in Optical Design Optimization Problems In our experience, optical designers often think orthogonal polynomials behave better during optical system optimizations. This is reported by empirical investigations [74][6]. However, since the only commonly used non-orthogonal polynomials are the monomials, we wonder whether it could be that monomials give rise, as a general rule, to slower convergence in optimization, not because they are not orthogonal but because of some other unrelated factor[7].

For a polynomial set to be useful in optical design, we can give a requirement: The first polynomial terms must be relevant to the problem at hand. We mean that they must allow the greatest minimization of the merit function in as few terms as possible. In most cases, this means that the first polynomial terms must correct the low order aberrations (which are usually the largest in design starting points)[8]. This is the case for all the polynomials presented here.

2.1.1.6 RMS Departure of Orthogonal Polynomials

The RMS departures in altitude $RMS_{\text{alt}}(z_{\text{poly}})$ or in gradient $RMS_{\text{grad}}(z_{\text{poly}})$ are often used as a simple manufacturability metrics for freeform optics at the optical design stage. For a general freeform surface without orthogonality properties, these can be computed by evaluating the altitude/gradient on a discrete uniform sampling of a sufficient number N_{points} of points over the surface, as in Eqs. 2.16 and 2.17.

[6] "The convergence speed is much higher for orthogonal polynomial sets than for non-orthogonal (Monomials),[...]"

[7] We sometimes encounter a misconception among optical designers that using orthogonal polynomials gives rise to separable optimization problems, in much the same way as fitting optical surfaces with orthogonal polynomials is a separable problem (see Section 2.1.1.7). This is emphatically *not* the case: for a general optical system (*eg* a TMA), the merit function is *not* separable with regards to the freeform coefficients variables defining the optical surfaces.

[8] There is no one-size-fits-all polynomial set, the performance will always be related to the problem at hand. Most of the presented polynomials are very adequate for correcting low order optical aberrations. However, in edge cases where the Merit Function (MF) is not related to aberration correction, or if very high order of aberrations were the optimization target, the presented polynomials could become inadequate.

$$RMS_{\text{alt}}(z_{\text{poly}}) \approx \sqrt{\frac{\sum_{i=1}^{N_{\text{points}}} z_{\text{poly}}^2(x_i, y_i)}{N_{\text{points}}}} \tag{2.16}$$

$$RMS_{\text{grad}}(z_{\text{poly}}) \approx \sqrt{\frac{\sum_{i=1}^{N_{\text{points}}} \left(\frac{\partial z_{\text{poly}}}{\partial x}(x_i, y_i)\right)^2 + \left(\frac{\partial z_{\text{poly}}}{\partial y}(x_i, y_i)\right)^2}{N_{\text{points}}}} \tag{2.17}$$

For the case of orthogonal polynomials with respect to an integral inner product over some spatial domain (unit disk here), computing the RMS quantities becomes exact and less computationally intensive, they are a function of the polynomials coefficients c_j:

- For polynomials orthogonal in altitude: Eq. 2.18.
- For polynomials orthogonal in gradient: Eq. 2.19.

$$\begin{aligned} RMS_{\text{alt}}(z_{\text{poly}}) &= \sqrt{\frac{1}{\pi} \int_{\theta=-\pi}^{\pi} \int_{\rho=0}^{1} \left(\sum_{j=0}^{j_{\max}} c_j P_j(\rho, \theta)\right)^2 \rho d\rho d\theta} \\ &= \sqrt{\sum_{j=0}^{j_{\max}} \left(\text{const}(j) \cdot c_j\right)^2} \end{aligned} \tag{2.18}$$

$$\begin{aligned} RMS_{\text{grad}}(z_{\text{poly}}) &= \sqrt{\frac{1}{\pi} \int_{\theta=-\pi}^{\pi} \int_{\rho=0}^{1} \overrightarrow{\nabla} \sum_{j=0}^{j_{\max}} c_j P_j(\rho, \theta) \cdot \overrightarrow{\nabla} \sum_{j=0}^{j_{\max}} c_j P_j(\rho, \theta) \rho d\rho d\theta} \\ &= \sqrt{\sum_{j=0}^{j_{\max}} \left(\text{const}(j) \cdot c_j\right)^2} \end{aligned} \tag{2.19}$$

2.1.1.7 Fitting Surfaces with Orthogonal Polynomials

We can approximate a continuous function $f(\rho, \theta)$ using a finite series of orthogonal polynomials P_j with coefficients c_j with $j \in [\![0 \dots j_{\max}]\!]$ resulting in an approximation $f_{\text{fit}}(\rho, \theta)$ (Eq. 2.20). If the RMS is taken in the usual sense in optics (in nanometers and unweighted spatially), we need orthogonal polynomials with respect to an integral of an altitude product over a spatial domain (such as Zernike polynomials).

$$f_{\text{fit}}(\rho, \theta) = \sum_{j=0}^{j_{\max}} c_j \cdot P_j(\rho, \theta) \tag{2.20}$$

The solution set to our minimization problem is the set of coefficients c_j that minimize a distance metric between f and f_{fit}. In most cases, we want to minimize the RMS altitude errors between the two functions (Eq. 2.21). For the sake of clarity, we work in the unit disk, but the case is adaptable to any support over which the polynomial basis is orthogonal.

$$\text{RMS}(f, f_{\text{fit}}) = \sqrt{\int_{\theta=-\pi}^{\pi} \int_{\rho=0}^{1} \left(\sum_{j=0}^{j_{\max}} c_j \cdot P_j(\rho, \theta) - f(\rho, \theta) \right)^2 \rho d\rho d\theta} \tag{2.21}$$

Now, using the polynomials orthogonality condition (Eq. 2.12) and the fact that f can be expressed in the polynomial basis P with coefficients c'_j (Eq. 2.22). We can deduce (with some steps omitted) Eq. 2.23.

$$f(\rho, \theta) = \sum_{j=0}^{\infty} c'_j \cdot P_j(\rho, \theta) \tag{2.22}$$

$$\text{RMS}(f, f_{\text{fit}}) = \pi \left(\sum_{j=0}^{j_{\max}} \text{const}(j)(c_j - c'_j)^2 + \sum_{j=j_{\max}+1}^{\infty} \text{const}(j)(c'_j)^2 \right) \tag{2.23}$$

We can easily see from Eq. 2.23 that the individual coefficients c_j will have an independent effect on $\text{RMS}(f, f_{\text{fit}})$. Thus, the minimization problem is separable and vastly easier to solve numerically than if it were non-separable.[9]

Please note the orthogonality is broken by:

- Not working on the support over which the polynomials are defined and orthogonal. For example working with a subaperture of unit-disk polynomials completely breaks the orthogonality. Likewise, extrapolating the polynomials outside of their support breaks the orthogonality.
- Working with an arbitrary discrete sampling of the surface described by $f(\rho, \theta)$, this is obviously always the case with measurement data. Gray [75] reports in Appendix III a particular discrete sampling that allows orthogonality for Zernike polynomials, as well as a method to find a Zernike polynomial expansion for a surface without the need for an iterative search process. Most of the time however, surfaces are fitted with a sufficient number of data points so that the orthogonality approximately holds[10].

On the subject of fitting a discrete sample of points with polynomials, we refer the reader to [76] for the importance of the sampling scheme.

[9] In a search problem that is so-called *separable*, each parameter can be optimized independently. For instance, say we seek the minimum of $f(c_1, c_2, c_3, c_4)$ (4 dimensional search problem). We can find the value of each parameter that minimizes $f(c_j)$ with all the others fixed at arbitrary values. The four optimal arguments also provide the minimum to $f(c_1, c_2, c_3, c_4)$ if f is separable with respect to its four arguments. Thus the 4D search problem is reduced to four much simpler 1D search problems.

[10] The integral in Eq. 2.12 becomes a finite sum over the sampling points.

2.1.2 Commonplace Polynomials

2.1.2.1 Zernike

Zernike polynomials $Z_n^m(\rho,\theta)$ are defined explicitely as the product of a *radial* component $R_n^m(\rho)$ and an *azimuthal* component $A_m(\theta)$ as in Eq. 2.24 [77].

$$Z_n^m(\rho,\theta) = R_n^m(\rho)\cdot A_m(\theta) \tag{2.24}$$

$$R_n^m(\rho) = \sum_{k=0}^{(n-|m|)/2} (-1)^k \frac{(n-k)!}{k!(\frac{n+m}{2}-k)!(\frac{n-m}{2}-k)!}\rho^{n-2k} \tag{2.24a}$$

$$A_m(\theta) = \begin{cases} \cos m\theta & m \geq 0 \\ \sin |m|\theta & m < 0 \end{cases} \tag{2.24b}$$

Please note the explicit expressions *cannot* be used for $n > 8$ approximately [78] on typical 64-bits computers performing floating-point arithmetic. The numerical error grows with n, *e.g.* in the computation of the ρ^n terms[11]. See Section 2.1.3.1 for a numerically robust evaluation strategy.

We give Fig. 2.7, a table of the first few terms of the Zernike polynomials in the *Standard* normalization and the OpticStudio indexing for the surface type *Zernike Standard Sag*. For more details on the Zernike polynomials, the different conventions, indexing, normalizations and properties, we refer the reader to the dedicated Appendix A.

2.1.2.2 XY (monomials)

So-called *XY polynomials*, *monomials* or *Extended Polynomials* (in OpticStudio) can be defined over a double indexing (p,q) as Eq. 2.25.

$$P_{p,q}^{\mathrm{XY}}(x,y) = x^p y^q \tag{2.25}$$

XY polynomials have no orthogonality property. It is our view that one should not dismiss these polynomials for this reason. Indeed, perfectly good optical systems were designed using XY polynomials [80, 81]. Additionally, their simple definition allows very clear communication. We plot the first few polynomials in Fig. 2.8.

We adopt the OpticStudio single indexing for *Extended Polynomials*. The terms are arranged in increasing order $n = p + q$ and inside each order in decreasing exponent p and increasing q. We start the indexing at $j = 0$ for the term X^1Y^0. To compute the single index j from (p,q), use Eq. 2.26. To compute the exponents (p,q) from j, use Eq. 2.27.

$$\begin{cases} n = p+q \\ j = \frac{n(n+1)}{2} - 1 + q \end{cases} \tag{2.26}$$

[11] The arithmetically inclined readers can refer to [79] on the subject.

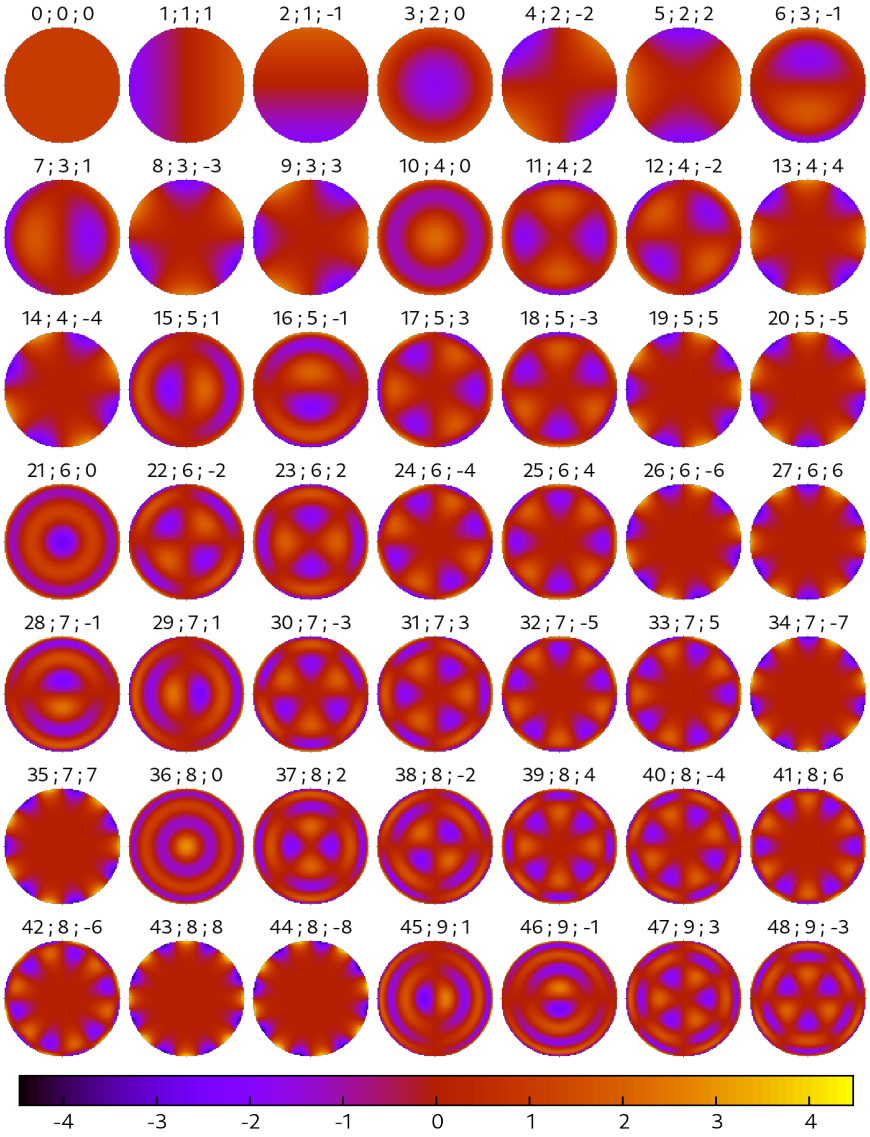


Figure 2.7: Table of the first few Zernike polynomials, as computed by our implementation using Andersen's work [78] and the OpticStudio "Zernike Standard Sag" indexing and conventions. For each polynomial, we give three numbers: j, n, m with j the Noll/OpticStudio single-index starting from 0 and n, m the corresponding radial and azimuthal orders respectively. The color scale is common to all the polynomials and the terms use the *standard* normalization.

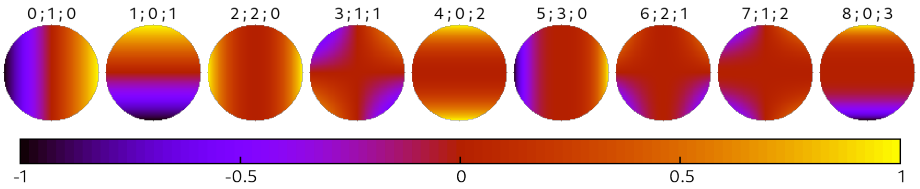


Figure 2.8: XY polynomials table. The indices on top of each map are $j;p;q$. j is the OpticStudio single index, p the x exponent and q the y exponent. Only the very first terms are represented here. The following terms have altitudes at the edge that is so high that maps with a linear colormap appear mostly flat.

$$\begin{cases} n = \left\lfloor \frac{-1+\sqrt{1+8(j+1)}}{2} \right\rfloor \\ p = n - j + \frac{n(n+1)}{2} - 1 \\ q = j - \frac{n(n+1)}{2} + 1 \end{cases} \tag{2.27}$$

2.1.2.3 Q-poly (Forbes)

The freeform polynomials commonly referred to as *Q-poly* are due to G.W. Forbes [82], building upon his previous work on the representation of rotationally symmetric aspheres [83]. Eq.2.2 of [82] gives the expression for a general surface with a spherical base shape employing Q-polys. It also uses a spherical projection factor Eq. 2.9. Ignoring the base shape and projection issues, we can express Q-polys as Eq. 2.28 (Eq.B.1 in [82]).

$$\begin{aligned} P^{\mathrm{Q}}(\rho,\theta) =& \rho^2(1-\rho^2)\sum_{n=0}^{n_{\max}} \mathrm{rot}_n Q_n^0(\rho^2) \\ &+ \sum_{m=1}^{m_{\max}} \rho^m \left(\cos m\theta \sum_{n=0}^{n_{\max}} a_n^m Q_n^m(\rho^2) + \sin m\theta \sum_{n=0}^{n_{\max}} b_n^m Q_n^m(\rho^2) \right) \end{aligned} \tag{2.28}$$

Notice how the formula can be broken down into:

- A rotationally symmetric component: $\rho^2(1-\rho^2)\sum_{n=0}^{n_{\max}} \mathrm{rot}_n Q_n^0(\rho^2)$
- A freeform component:
  $\sum_{m=1}^{m_{\max}} \rho^m \left(\cos m\theta \sum_{n=0}^{n_{\max}} a_n^m Q_n^m(\rho^2) + \sin m\theta \sum_{n=0}^{n_{\max}} b_n^m Q_n^m(\rho^2)\right)$

The rotationally symmetric component is *exactly* the aspheric polynomials Q^{bfs} shown in [83]. The freeform component has a part in cos and a part in sin much like Zernike polynomials. Explicit formulae for the terms $Q_n^m(\rho^2)$ are given in Fig.3 of [82], although they quickly become numerically intractable for higher orders of n and m.

The available parameters for this surface are:

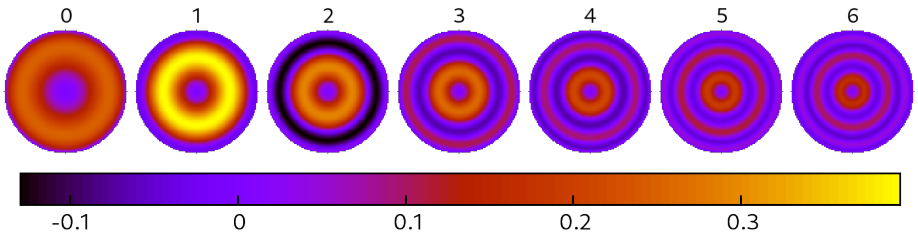


Figure 2.9: Table for the rotationally symmetric terms $\rho^2(1-\rho^2)Q_n^0(\rho^2)$ for $n = [0, 6]$. The color scale is the same for all terms. Notice how the extremum decreases along with order n. The radial spatial frequency also increases with order. A cross-section plot of these terms can be found in Fig.2 of [83].

- rot_n : Coefficients for rotationally symmetric terms. A table for these terms is shown in Fig. 2.9.
- a_n^m : Coefficients for the *cos* component of the freeform part. A table is shown in Fig. 2.10.
- b_n^m : Coefficients for the *sin* component of the freeform part. A table is shown in Fig. 2.11 for the sake of completeness.

All these coefficients could be arranged in a single-indexed table, much in the way Zernike polynomials coefficients are.

The Q-polys P^Q of Eq. 2.28 are *slope orthonormal* (see Eq. 2.13) over the unit-disk, but not orthogonal in altitude. This enables the computation of a manufacturing criterion of RMS departure in gradient in the efficient, exact and straightforward way highlighted in Eq. 2.19.

A robust and efficient way of computing the Q-polynomials surfaces and their derivatives to any order is shown by Forbes. As it happens, sums can be computed directly in Eq. 2.28 using Clenshaw's method (Section 2.1.3.1). This allows robust and efficient evaluation. See [82, 84] for the application of Clenshaw's method to Q-polynomials. All the tables of Q-polynomials we show in the present work are computed using this method.

2.1.2.4 Legendre

Legendre polynomials, in a mathematical context, refer to 1D polynomials, an explicit equation of which is Eq. 2.29 (one of many such formulae) [85], with $n = [0, n_{\max}]$.

$$L_n(x) = \frac{1}{2^n} \sum_{k=0}^{\lfloor n/2 \rfloor} (-1)^k \binom{n}{k} \binom{2n-2k}{n} x^{n-2k} \tag{2.29}$$

For a more robust implementation, the recurrence relation in Eq. 2.30 [85] is indicated. Start with the first terms $L_0 = 1$, $L_1 = x$. Better still, the weighted

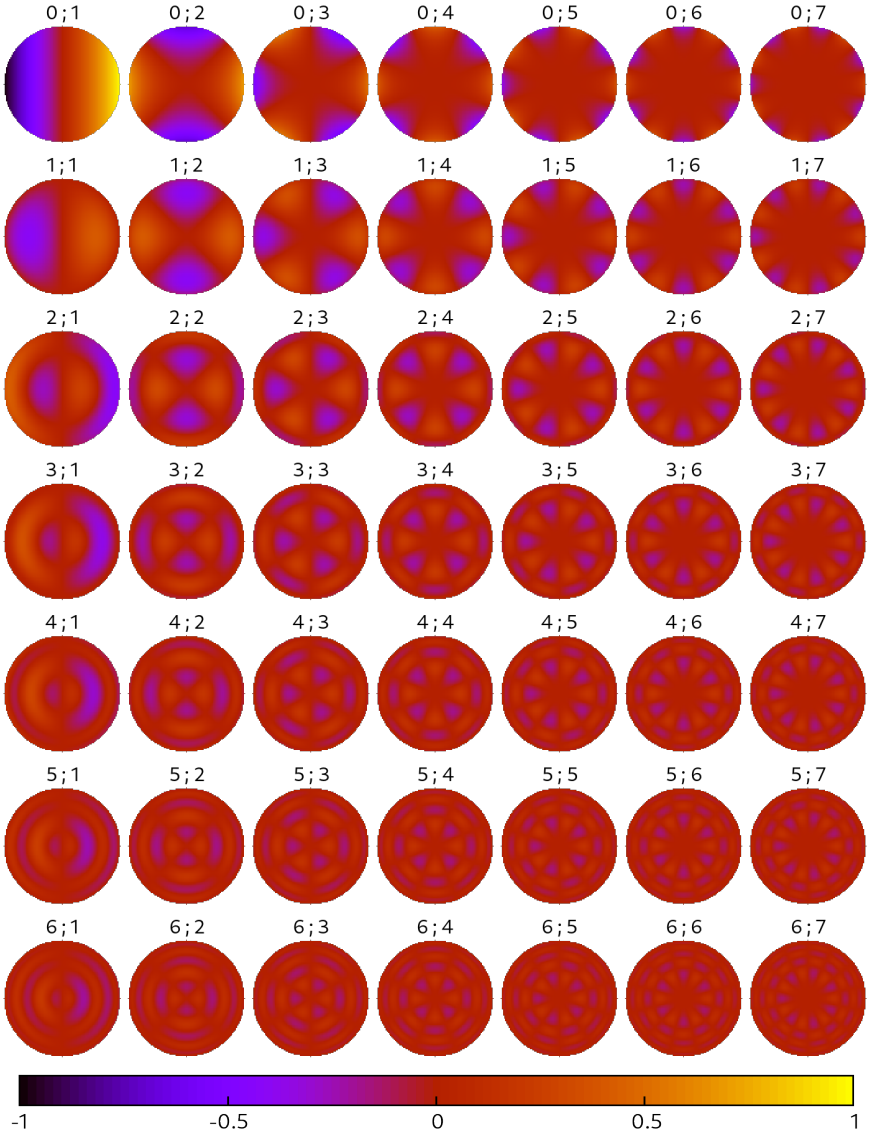


Figure 2.10: Table for the freeform cos terms $\rho^m \cos m\theta Q_n^m(\rho^2)$ for $n = [0, 6]$, $m = [1, 7]$. The color scale is the same for all terms. The indices written on top of each map are $n; m$. Notice how the extremum decreases along with orders n and m. The radial spatial frequency increases with n. The angular spatial frequency increases with m.

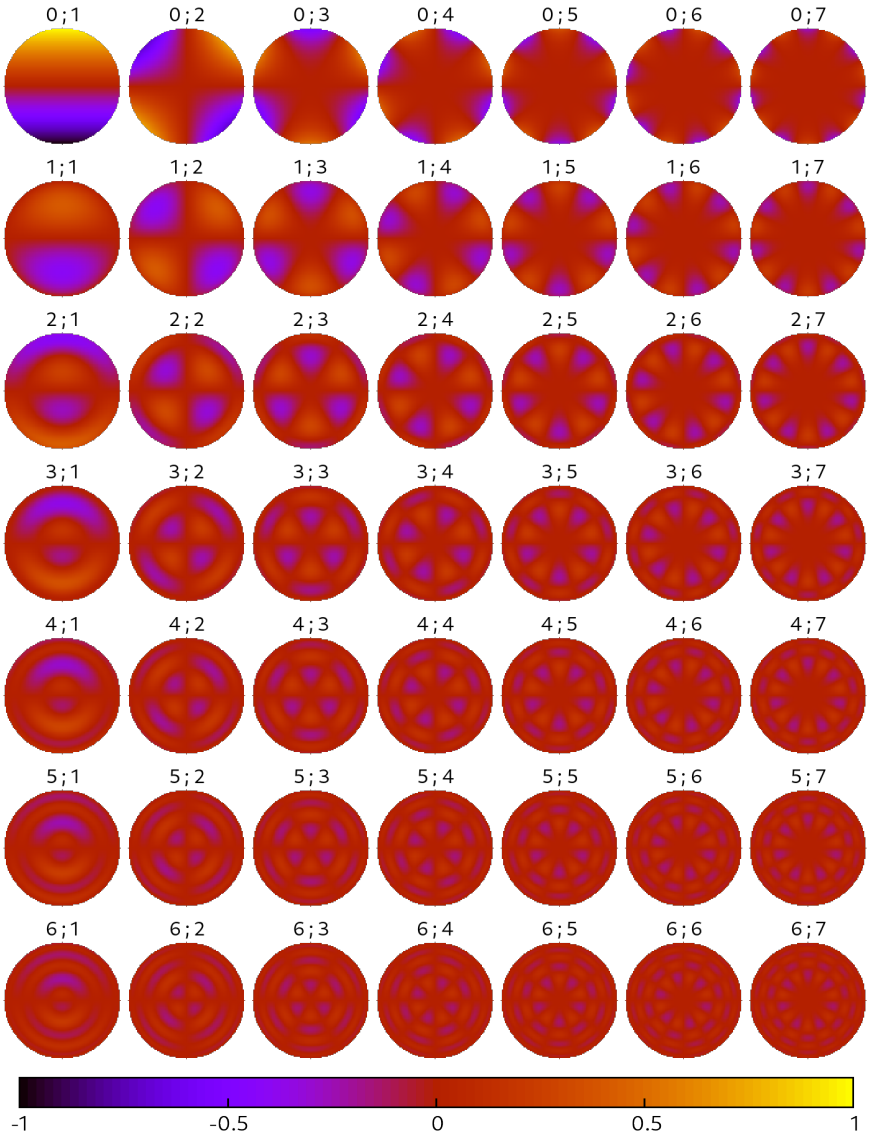


Figure 2.11: Table for the freeform sin terms $\rho^m \sin m\theta Q_n^m(\rho^2)$.

sum of Legendre terms and their derivatives can be computed directly using the general Clenshaw method (Section 2.1.3.1).

$$(1+n)L_{n+1}(x) - (2n+1)xL_n(x) + nL_{n-1}(x) = 0 \tag{2.30}$$

From the 1D Legendre polynomials, we can build a 2D polynomial for optics via a simple product in x and y, see Eq. 2.31 [67].

$$P^L_{m,n}(x,y) = L_m(x)L_n(y) \tag{2.31}$$

We then have a double-indexed table of coefficients $c_{m,n}$. We represent the first few 2D Legendre polynomial terms in Fig. 2.12.

The 1D Legendre polynomials are orthogonal over $[-1,1]$ [85], see Eq. 2.32.

$$\int_{x=-1}^{1} L_n(x) \cdot L_{n'}(x)dx = \frac{2\delta_{n,n'}}{2n+1} \tag{2.32}$$

It can further be shown that the 2D version is orthogonal on the unit-square $x = [-1,1]$, $y = [-1,1]$, using the separation along the two coordinates (x,y) and the 1D orthogonality property, see Eq. 2.33.

$$\begin{aligned}\langle P^L_{m,n}, P^L_{m',n'}\rangle &= \int_{y=-1}^{1}\int_{x=-1}^{1} L_m(x)L_n(y) \cdot L_{m'}(x)L_{n'}(y)dxdy \\ &= \frac{4\delta_{m,m'}\delta_{n,n'}}{(2m+1)(2n+1)}\end{aligned} \tag{2.33}$$

2.1.2.5 Chebyshev

As with Legendre polynomials, Chebyshev polynomials refer to 1D polynomials which we multiply along the two cartesian directions for use in optics. An explicit expression for Chebyshev polynomials of the first kind, in the interval $[-1,1]$ is given by Eq. 2.34 [86].

$$C^{1\text{st}}_n(x) = \cos(n \arccos(x)) \text{ if } |x| \leq 1 \tag{2.34}$$

The first kind is orthogonal on $[-1,1]$ with respect to a non-unit weight function, as seen in Eq. 2.35[12] [86].

$$\int_{-1}^{1} \frac{C^{1\text{st}}_n(x) \cdot C^{1\text{st}}_{n'}(x)}{\sqrt{1-x^2}}dx = \begin{cases} \frac{\pi\delta_{n,n'}}{2} & n \neq 0 \text{ and } n' \neq 0 \\ \pi & n = n' = 0 \end{cases} \tag{2.35}$$

[12] Note this is an improper integral.

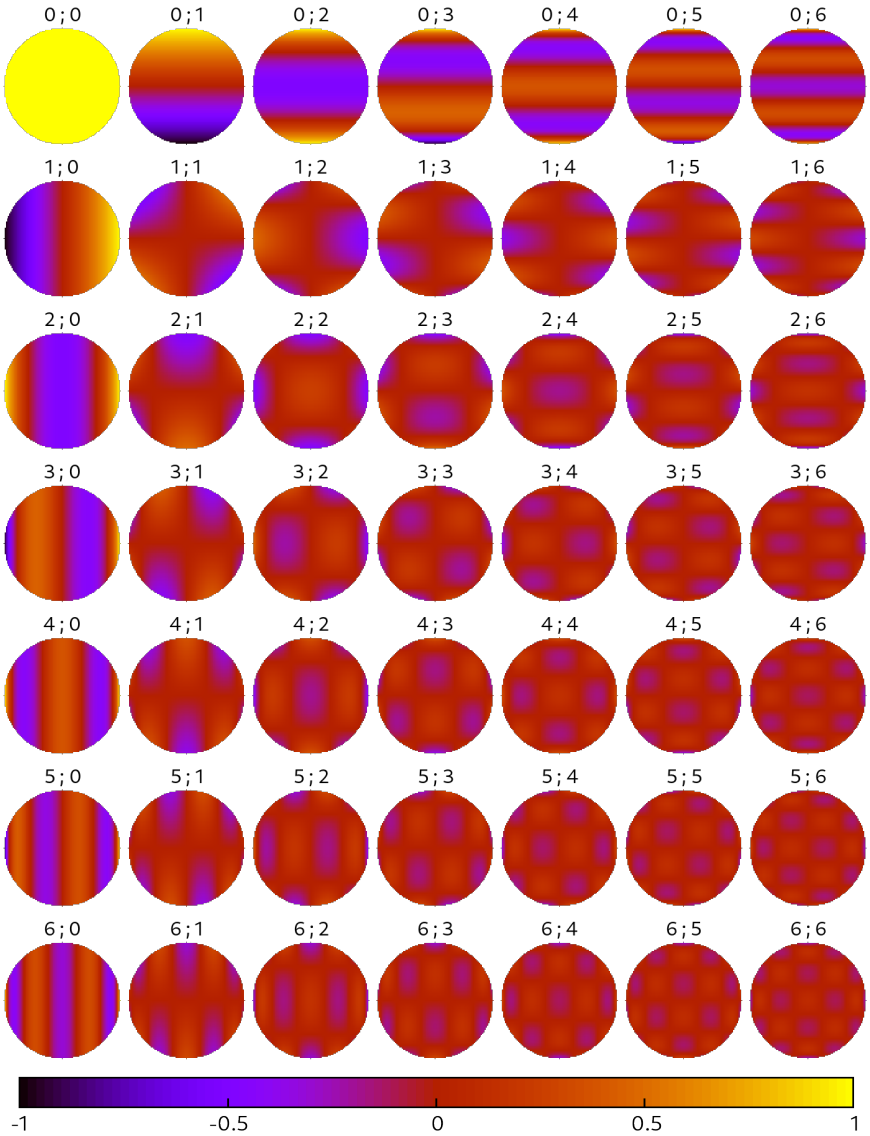


Figure 2.12: Table of Legendre 2D polynomials. The indices on top of each map are (m, n).

We can build a 2D Chebyshev polynomials of the first kind via a simple product shown in Eq. 2.36. The first few terms are shown in Fig. 2.13. It is straightforward to show these 2D polynomials are orthogonal on the unit square with respect to some non-unit weight function, see Eq. 2.37.

$$P_{m,n}^{C^{1\text{st}}}(x,y) = C_m^{1\text{st}}(x) C_n^{1\text{st}}(y) \tag{2.36}$$

$$\begin{aligned}
\langle P_{m,n}^{C^{1\text{st}}}, P_{m',n'}^{C^{1\text{st}}} \rangle &= \int_{y=-1}^{1} \int_{x=-1}^{1} \frac{C_m^{1\text{st}}(x) C_n^{1\text{st}}(y) \cdot C_{m'}^{1\text{st}}(x) C_{n'}^{1\text{st}}(y)}{\sqrt{(1-x^2)(1-y^2)}} dxdy \\
&= \begin{cases}
\frac{\pi^2 \delta_{m,m'} \delta_{n,n'}}{4} & \{m, m', n, n'\} \in \mathbb{N}^* \\
\frac{\pi^2 \delta_{m,m'}}{2} & n = n' = 0, \{m, m'\} \in \mathbb{N}^* \\
\frac{\pi^2 \delta_{n,n'}}{2} & m = m' = 0, \{n, n'\} \in \mathbb{N}^* \\
\pi^2 & m = m' = n = n' = 0
\end{cases}
\end{aligned} \tag{2.37}$$

A possible explicit expression for Chebyshev polynomials of the second kind is given by Eq. 2.38 [87].

$$C_n^{2\text{nd}}(x) = \sum_{m=0}^{\lfloor n/2 \rfloor} \binom{n+1}{2m+1} x^{n-2m} (x^2-1)^m \tag{2.38}$$

Chebyshev polynomials of the second kind are orthogonal on $[-1, 1]$ with respect to a different non-unit weight function, see Eq. 2.39.

$$\int_{-1}^{1} C_n^{2\text{nd}}(x) \cdot C_{n'}^{2\text{nd}}(x) \sqrt{1-x^2} dx = \frac{\pi \delta_{n,n'}}{2} \tag{2.39}$$

A 2D version of Chebyshev polynomials can be constructed as in Eq. 2.40. The first few terms are shown in Fig. 2.14. These polynomials are orthogonal on the unit square with respect to a non-unit weight function, as Eq. 2.41 shows.

$$P_{m,n}^{C^{2\text{nd}}}(x,y) = C_m^{2\text{nd}}(x) C_n^{2\text{nd}}(y) \tag{2.40}$$

$$\begin{aligned}
&\langle P_{m,n}^{C^{2\text{nd}}}, P_{m',n'}^{C^{2\text{nd}}} \rangle \\
&= \int_{y=-1}^{1} \int_{x=-1}^{1} C_m^{2\text{nd}}(x) C_n^{2\text{nd}}(y) \cdot C_{m'}^{2\text{nd}}(x) C_{n'}^{2\text{nd}}(y) \sqrt{(1-x^2)(1-y^2)} dxdy \\
&= \frac{\pi^2}{4} \delta_{m,m'} \delta_{n,n'}
\end{aligned} \tag{2.41}$$

Another definition for both first kind and second kind is with the recurrence relation in Eq. 2.42 [86] with the starting terms given in Eq. 2.43. As always with orthogonal polynomials, Clenshaw (Section 2.1.3.1) applies.

$$C_{n+1}(x) = 2xC_n(x) - C_{n-1}(x) \tag{2.42}$$

$$\begin{cases} C_0^{\text{1st}}(x) = 1; C_1^{\text{1st}}(x) = x & \text{first kind} \\ C_0^{\text{2nd}}(x) = 1; C_1^{\text{2nd}}(x) = 2x & \text{second kind} \end{cases} \tag{2.43}$$

2.1.2.6 Bernstein

Researchers at LAM [88, 89] have shown the successful application of Bernstein polynomials to the description of surfaces for optical design. They have described 2D polynomials and a further orthogonalization method. We will show the non-orthogonal version here for the sake of simplicity. Over the unit-square, we can define the 2D Bernstein polynomials as Eq. 2.44 with a double-indexing scheme (i, j) and respective maximum orders (n, m).

$$\begin{aligned} &P^B_{i,j,n,m}(x, y) = \\ &\binom{n}{i} \left(\frac{1+x}{2}\right)^i \left(\frac{1-x}{2}\right)^{n-i} \binom{m}{j} \left(\frac{1+y}{2}\right)^j \left(\frac{1-y}{2}\right)^{m-j} \end{aligned} \tag{2.44}$$

With the usual binomial coefficient:

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} \tag{2.45}$$

The Bernstein polynomials have the following properties:

- The highest orders (n, m) have to be defined in advance.
- The Bernstein polynomials through the maximum order form a *partition of unity*. This means that everywhere on the surface: $\sum_{i=0}^{n} \sum_{j=0}^{m} P^B_{i,j,n,m}(x, y) = 1$
- Each term is positive everywhere on the surface: $P^B_{i,j,n,m}(x, y) \geq 0$

We give a table of Bernstein polynomial terms for $(n, m) = (6, 6)$ Fig. 2.15. Each term is mostly a local deformation of the surface, the indices (i, j) move the center of the deformation along the two cartesian dimensions. We refer the reader to [88] for the derivation through singular value decomposition of a set of orthogonal terms, which the authors note should be more suited to optical design.

2.1.3 Generalizations and Computerized Implementation

The mathematics and algorithmic literature provide powerful shortcuts and points of view for the implementation of polynomials.

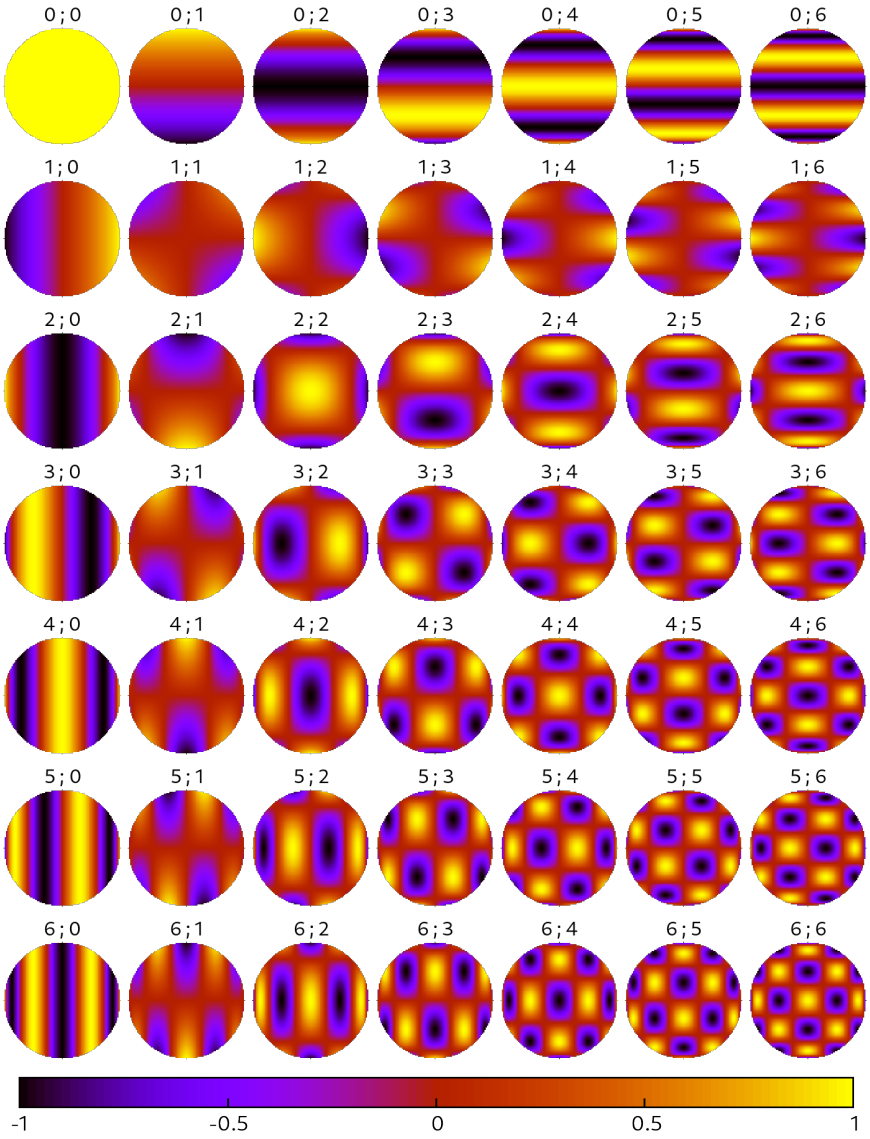


Figure 2.13: Table of 2D Chebyshev polynomials of the first kind $P_{m,n}^{C^{1\text{st}}}(x,y)$. The indices on top of each map are (m,n).

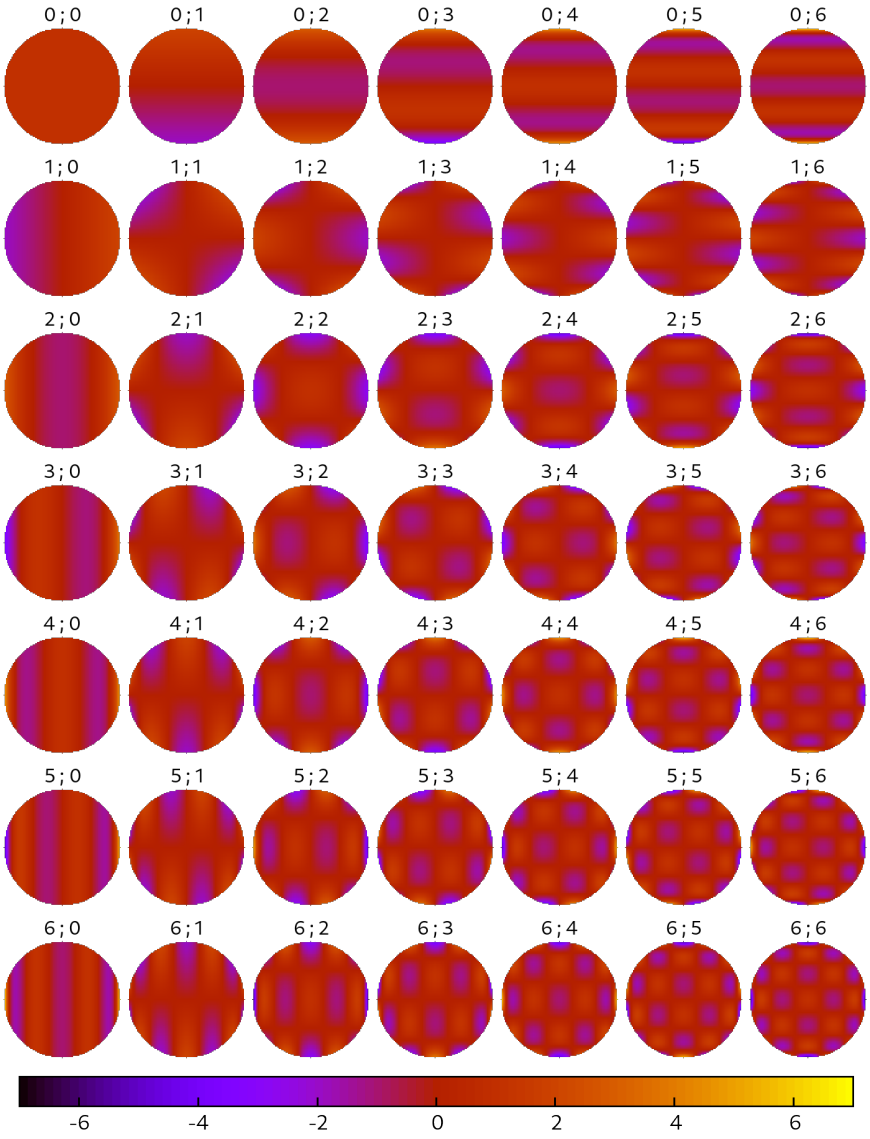


Figure 2.14: Table of 2D Chebyshev polynomials of the second kind $P_{m,n}^{C^{\mathrm{2nd}}}(x,y)$. The indices on top of each map are (m,n).

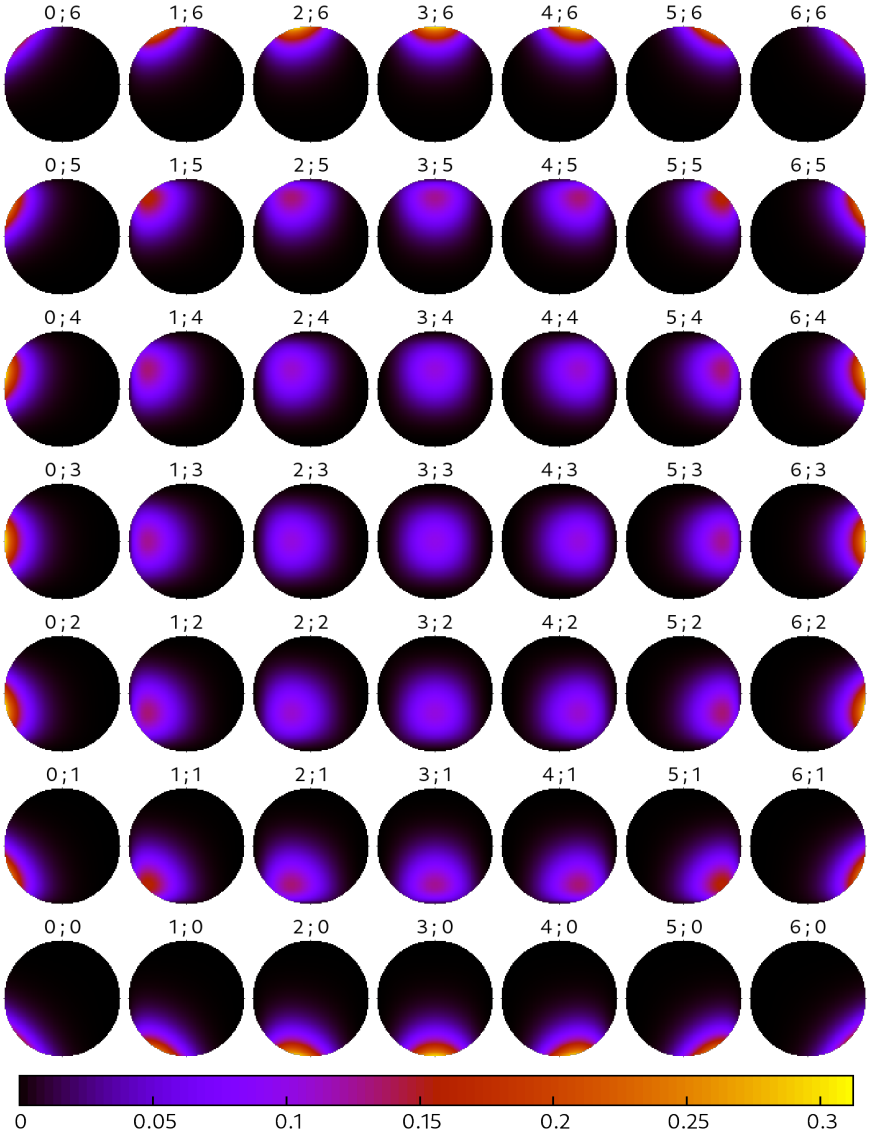


Figure 2.15: Berstein polynomials table for $(n, m) = (6, 6)$. The indices we give on top of each map are (i, j). Notice how each map is a local deformation of the surface.

2.1.3.1 General implementation for orthogonal polynomials: Clenshaw's method

As noted by Forbes [90], *all orthogonal polynomials* (1D) with terms $P_n(x)$ can be expressed with the aid of the recurrence relation of the type given in Eq. 2.46.

$$P_{n+1}(x) = (A_n + B_n x) \cdot P_n(x) - C_n \cdot P_{n-1}(x) \tag{2.46}$$

$$P_0(x) = 1 \tag{2.46a}$$

$$P_1(x) = A_0 + B_0 x \tag{2.46b}$$

What follows is true for all polynomials that obey Eq. 2.46. Not all polynomials can be expressed in this way, but *all* orthogonal polynomials can, as can a few other non-orthogonal polynomials, *eg* monomials with $A_n = C_n = 0$ and $B_n = 1$.

Defining the particular polynomials is then just a matter of setting A_n, B_n and C_n. A table for most classical orthogonal polynomials is found online in the NIST DLMF (18.9(i)) [91] (careful, A_n and B_n are switched). This general recurrence formula is already more numerically robust than most explicit expressions, but there is more. For *any* orthogonal polynomial, one can then evaluate finite sums of weighted polynomials and their derivatives using an incredibly simple and efficient relation we owe to Clenshaw, and highlighted in [90]. Evaluating the altitude and first derivatives is all that is needed for simple raytracing, so the implications for optical design code are self-evident.

For the evaluation of a weighted sum $S(x)$ (altitude in optical design) (see Eq. 2.47) of polynomial terms P_m (polynomials that obey the recurrence relation Eq. 2.46) at point x, with coefficients c_m, we can introduce the quantity α_n which follows the (descending) recurrence relation given in Eq. 2.48 [90].

$$S(x) = \sum_{m=0}^{M} c_m \cdot P_m(x) \tag{2.47}$$

$$\alpha_n = c_n + (A_n + B_n x) \cdot \alpha_{n+1} - C_{n+1} \cdot \alpha_{n+2} \tag{2.48}$$

$$\alpha_M = c_M \tag{2.48a}$$

$$\alpha_{M-1} = c_{M-1} + (A_{M-1} + B_{M-1} x) \cdot c_M \tag{2.48b}$$

The sum $S(x)$ can then be obtained through the exceedingly simple Eq. 2.49.

$$S(x) = \alpha_0(x) \tag{2.49}$$

Similarly, derivatives of order j, $S^{(j)}(x)$ (Eq. 2.50) are obtained by introducing $\alpha_n^{(j)}$ obeying the recurrence relation in Eq. 2.51. The result is given by Eq. 2.52.

$$S^{(j)}(x) = \sum_{m=0}^{M} c_m \cdot P_m^{(j)}(x) \tag{2.50}$$

$$\alpha_n^{(j)} = jB_n \cdot \alpha_{n+1}^{(j-1)} + (A_n + B_n x) \cdot \alpha_{n+1}^{(j)} - C_{n+1} \cdot \alpha_{n+2}^{(j)} \tag{2.51}$$

$$\alpha_{M-j+1}^{(j)} = 0 \tag{2.51a}$$

$$\alpha_{M-j}^{(j)} = jB_{M-j} \cdot \alpha_{M-j+1}^{(j-1)} \tag{2.51b}$$

$$S^{(j)}(x) = \alpha_0^{(j)}(x) \tag{2.52}$$

These relations are given in the 1D case, but are applicable to 2D polynomials that are defined using radial and azimuthal terms: the application to Zernike polynomials is detailed in [90], to Q-polys in [82, 83], note this is always the same general method. Also note that during the implementation of this method, some quantities (*eg* A_n, B_n, C_n) that are independent from x can be precomputed up to some maximum polynomial order, which further speeds up the computation.

We think it necessary to insist on the relevance of these relations. This is in *most* cases the best (quickest and most accurate up to high polynomial orders) way to compute *any* surfaces described by orthogonal polynomials (plus any other polynomials that obey the three-term recurrence in Eq. 2.46) and their derivatives in optics. It is moreover astonishingly simple and general. The subject of Clenshaw's method is also detailed in Numerical Recipes [92], with interesting discussions on the numerical stability of the recurrences used. A similar shortcut is available for changes of basis between polynomials that obey Eq. 2.46, owed to Salzer and discussed by Forbes [90].

A perhaps less powerful generalization for the presented polynomials is to realise that many can be directly linked to Jacobi polynomials (18.7 [91]). This is true of Chebyshev polynomials of all kinds, Legendre, Zernike [90], Q-poly [82].

2.1.3.2 Numerical methods for polynomials evaluation

When Clenshaw's method is not applicable, there are a few well-known methods that can be of interest when evaluating weighted sums of polynomials and their derivatives.

When the coefficients in front of polynomials are "ill-conditioned" (a mix of very large and very small coefficients), the sum will suffer from numerical inaccuracy[13]. To perform the sum while compensating for inaccuracies, one can use the Kahan (also Kahan-Babuska-Neumaier) summation algorithm [93].

[13] The reader can try evaluating the following sum in any programming language performing simple float arithmetics: $1.0 + 1\text{E}16 - 1\text{E}16 - 0.5$. The correct result is blatantly 0.5. However, if you tell your computer to sum the terms in that order, performing just single or double float arithmetics, the result will be -0.5. The first 1.0 gets squashed by 1E16.

For reference, several well-known evaluation schemes exist for monomials. Horner's scheme (see 5.1 in [92]) is a particular case of Clenshaw's method. Estrin's scheme [94] (also 5.1 [92]) allows the evaluation of another factorisation of monomials.

Additionally, since optical researchers will often want at least the first derivatives of polynomial surfaces in addition to the altitude, it should be noted that it is often possible to compute both altitude and derivatives simultaneously in less time than if they were evaluated independently [92].

2.1.3.3 Methods to generate new polynomials for optics

There is no reason to doubt other polynomials could be useful for the representation of freeform optics. There are several ways one could go about finding new polynomials.

First, one can perform a Gram-Schmidt orthogonalization of an existing polynomial basis, using a different integration support and/or a different weight function. This is what Broemel applied to the Zernike polynomials to create the so-called A-polynomials [67].

Second, a Singular Value Decomposition (SVD) can be performed on a polynomial basis to extract eigenmodes. In this way, we can construct a new polynomial basis of terms with increasing spatial frequency. The first terms are then likely to be relevant for freeform surface description. This is a means of obtaining a basis relevant for optics from a not so useful initial basis. This is applied to Bernstein polynomials in [88].

The third way is simply to find existing polynomials in the mathematics and engineering literature and apply them to optics. Most polynomial bases have increasing spatial frequency with the index of terms, making them likely to be relevant for optics. If not, Gram-Schmidt and/or SVD can be used, as described above. One can for example try playing with the parameters of Jacobi polynomials [95] and test the results in optical systems.

A fourth method would be to use the general definition given in Eq. 2.46 and feed A_n, B_n, C_n functions to generate new polynomials. A test of recurrence stability and eventually of orthogonality should be implemented. Defining a surface type in this way for optical design raytracing would be a worthwhile investigation, the A_n, B_n, C_n could be functions defined by the user that contain parameters exposed to the optimizer. We would then have a surface with a dynamically changing polynomials definition. The surface would be able to generate any orthogonal polynomials and some others, giving the optical designer enormous freedom.

Concerning this last method, we refer the reader to the Favard-Shohat theorem [96], which allows generating *only* (but not *all*) univariate orthogonal polynomials via a restricted version of Eq. 2.46.

The question of going from 1D to 2D polynomials remains however. Some commonly employed methods are: performing a product of two 1D terms along the cartesian coordinates (such as in Legendre, Bernstein, Chebyshev) ; evaluating the polynomials along the radial direction and multiplying by an azimuthal

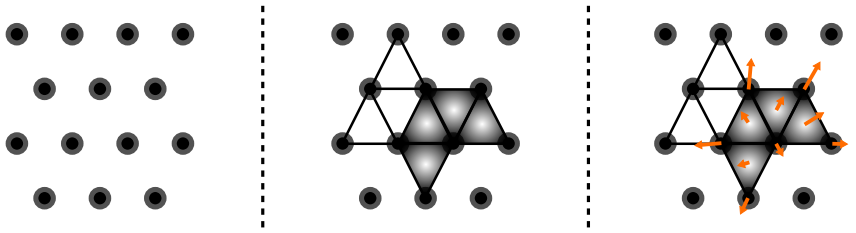


Figure 2.16: Illustration of different levels of sophistication of point-based representations. From left to right, the first representation is just a discrete sampling of altitudes. The second illustrates interpolation schemes between points, the altitude of points outside the sampling can be estimated. The third takes additionally into account a discrete sampling of normal vectors as well as an interpolation strategy for these.

trigonometric coefficient (Zernike and Q-poly). Both these methods seem restrictive. An example of going from 1D to 2D for Hermite and Laguerre polynomials while maintaining generality is shown in [97]. It would be better to generate 2D polynomials directly using some multivariate analogue to the three-term recurrence, the Favard-Shohat theorem and to Clenshaw's algorithm. As it happens, the three-term recurrence relation and Clenshaw's algorithm do generalize to polynomials of n (hence 2) variables [98, 99]. The n-dimensional generalization of the Favard theorem also exists [98]. We believe the investigation into the application to the generation and computation for the polynomials of optics would be fruitful[14].

2.2 Point-based Representations

Freeform surfaces may also be represented using *point-based* schemes. Broadly speaking, these are all the representations using grids (regular or not) of points in altitude, optionally with interpolation schemes (splines, patches etc.) or taking normal vectors into account. These different concepts are crudely illustrated in Fig. 2.16. These representations are usually found in the Computer-Aided Design (CAD) industry, or as measurement results. The high number of DOF of these representations (one or more for each point) makes them, in most cases, unsuitable for typical freeform optical design[15].

2.2.1 Point clouds

The simplest point-based representation is simply the point cloud. The continuous surface is sampled on a discrete grid (regular or not). This representation requires the most data points, which can become cumbersome, but is widely compatible and simple.

The number of data points requirement can be driven by:

[14] We plan on investigating the matter at a further date

[15] However, we note that OpticStudio, since a recent date, incorporates a freeform representation that is an interpolated point grid accessible to the optimization engine [100].

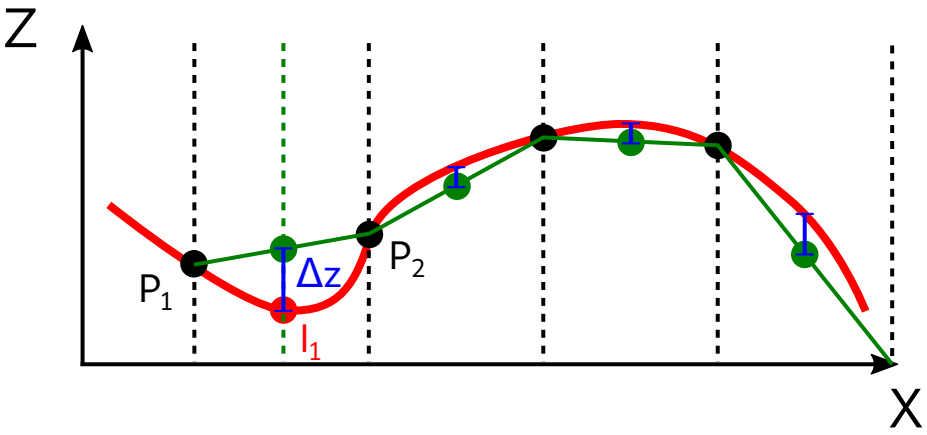


Figure 2.17: Simple estimation method of the error introduced by a discrete point-cloud sampling in the case of a linear interpolation. The red curve represents the continuous nominal surface. The black dots are the points in the point-cloud representation. The green line is the linearly interpolated surface from the point-cloud representation. An error Δz is evaluated in the middle of each interpolated segment along pairs of adjacent points on the grid. These individual quantities can be summed into a useful error metric.

- The mechanical size of the optical clear aperture on the surface.
- The altitude error requirements.
- The lateral spatial resolution of the manufacturing machine.

A simple way of estimating the error that is made when converting from a continuous representation to a point cloud is illustrated in Fig. 2.17. A (arguably worst case) linear interpolation is assumed and the error between pairs of adjacent points on the grid is evaluated.

We can estimate the error introduced via the local curvature $1/r$ of the optical surface (for surfaces with curvature variations of spatial frequency smaller than the sampling frequency and reasonable slopes). Taking r to be the local radius of curvature, approximately the same at points P_1, P_2 and I_1 (on Fig. 2.17), we can approximate the error introduced by the discrete sampling with Eq. 2.53, with a sampling step of δ_s.

$$\Delta z \approx \frac{(\delta_s)^2}{8r} \tag{2.53}$$

It follows that a good rule of thumb for creating a dynamic discrete sampling scheme for freeform surfaces is to densify the samples where high local curvatures occur. Alternatively, an iterative sampling scheme using the error metric Δe can be devised. Dynamic samplings diminish the number of required samples while maintaining an arbitrarily low error over the surface. However, they are less compatible with common software than the regular cartesian grid point clouds.

2.2.2 Interpolation, normal vectors and splines

The choice of interpolation for evaluating the altitude of the optical surface between sample points in a point-cloud greatly influences the introduced error. The derivatives can be included in the data for a better interpolated representation.

Some usual surface interpolation methods are (particular cases of Hermite interpolation):

- Bilinear: Generalization of linear interpolation to 2D.
- Bicubic: Generalization of cubic interpolation to 2D.

Alternatively, optical surfaces may be represented using B-spline-based schemes. A practical difference between Hermite interpolation and splines is that splines do not necessarily pass through the data points (named *control points*). Their application in optical design is investigated in [101–103].

Other methods include polygon patches. They generally allow the interpolation of not only the altitude but also the normal vector (which is required for raytracing). An example of this type of surface representation is the *Nagata patch* [104].

2.3 Radial Basis Functions and Local Series Expansions

This last surface representation family deals with sums of decentered local functions, such as Radial Basis Function (RBF)[16]. A RBF is a function the value of which depends only on the distance between the points in the input space and a center [106]. In the case of optical surfaces, these are all the rotationally symmetric functions around some center (x_c, y_c) on the surface. For example, 2D gaussian functions are RBF, an application to optical design with a sum of 2D gaussians is found in [107]. Optical surfaces can be approximated up to an arbitrarily small error with a growing number of local functions.

RBF can be used in combination with polynomial representations in order to approximate (or indeed define) surfaces more efficiently than with polynomials or RBF alone [108]. It is reported that some complex surface shapes can be represented with only a few dozen polynomial terms and RBF. Another proposed hybrid method between polynomials and RBF (conics in this case) is in [109].

On a related topic, [110] investigated the ability of Zernike polynomials to approximate a local gaussian deformation. Another investigation of this type with Zernike and Q polynomials is carried out in [111].

[16] RBF were an active research area in the 1990s for the approximation of multivariate functions. [105] is a widely cited reference on the subject.

2.4 General remarks

2.4.1 An edge case: Hyperhemispheres

Optical designers might wonder how to extend freeform optical surfaces definitions into the realm of near-hemispherical and hyperhemispherical diopters (*eg* in wide-angle lenses and microscopy). The *projection factor* can only mitigate problems for moderately curved surfaces. Near the equator of a sphere or into the second hemisphere, we encounter a fundamental coordinates issue: Most freeform surfaces presented here are defined using a $z = f(x, y)$ definition. It breaks down completely in the case of hyperhemispheres.

A crude solution to this problem is to use the usual $f(x, y)$ definition, as is, in a spherical coordinate system. The $u - v$ system or the so-called *double-pole coordinate system* [112] appear adequate. We give Eq. 2.54 and Eq. 2.55, adapted from [112] for the conversion of double-pole coordinates $(\alpha, \beta, r)_R$ to and from cartesian coordinates (x, y, z).

$$\begin{cases} r = \sqrt{x^2 + y^2 + z^2} - R \\ \alpha = \operatorname{atan}\left(\frac{x}{z+R+r}\right) \\ \beta = \operatorname{atan}\left(\frac{y}{z+R+r}\right) \end{cases} \tag{2.54}$$

$$\begin{cases} x = (R + r)(1 + \delta)\tan(\alpha) \\ y = (R + r)(1 + \delta)\tan(\beta) \\ z = (R + r)\delta \end{cases} \tag{2.55}$$

With:

$$\delta = \frac{1 - \tan^2(\alpha) - \tan^2(\beta)}{1 + \tan^2(\alpha) + \tan^2(\beta)} \tag{2.56}$$

As a naive first approach, we can treat the coordinates (α, β) exactly as though they were cartesian coordinates, and compute $r = f(\alpha, \beta)$, the radial deviation from a given sphere of radius R.

2.4.2 A variety of representations and a variety of use cases

Let us look along the other dimension (looking across use cases rather than representation families), we give typical tasks of optical engineering and how each freeform representation family (polynomial, point-based or RBF) relates to them.

2.4.2.1 In optical design minimization problems

Optical design traditionally makes use of polynomial representations for the good reason that they provide very relevant DOF for minimizing most merit

functions in a typical minimization problem. Indeed, a grid-based representation with thousands of DOF will likely converge very slowly towards a solution, if at all, given the structure of usual search algorithms and the curse of dimensionality. However, the optimization on sparse, interpolated point clouds is tractable and useful (see the *TrueFreeform* surface in OpticStudio). The RBF sums or hybrid representations also provide a viable alternative, given that the required number of DOF is manageable (as seen in [108]).

2.4.2.2 In direct design

In the context of *direct-design* or *construction* methods [113–115], we compute a point-based representation: the intersection and slope of rays on an optimal surface. As a subsequent step, a polynomial fit is sometimes performed.

2.4.2.3 Communicating the surface to a manufacturer

In theory, polynomial representation are ideal for the purpose of communicating the shape of an optical surface to be manufactured. In practice however, polynomials give rise to a considerable risk of misunderstanding between the optical designer and the manufacturer. Exotic polynomials will not be compatible with the manufacturer's software, and even in the case of the widely used Zernike polynomials, the variety of conventions (see Appendix A) complicates things considerably. It is advisable to provide redundant representations of different types:

- A **polynomial representation** (usually Zernike) with sufficient information on its definition, conventions *etc.* It might also be important to determine a set of coefficients and base shape that will minimize the freeform departure (*eg* transfer the optical power from the freeform definition to the base sphere).
- A **point cloud**. Where applicable, choose a sufficient density so that the error is acceptable when the surface is interpolated in any way possible.
- A **CAD model** of the optical surface plus the surrounding mechanical part. This is to make sure the position of the optical surface with regards to the part is correct, and what positive altitude values mean. Keep in mind that CAD models can suffer from significant surface shape errors and may not be suitable for the representation of the optical surface.

2.4.2.4 In surface metrology

Across measurement methods, metrology results are usually given in point-based representations. The raw results will often additionally be fitted with orthogonal polynomials (in our experience overwhelmingly Zernike), for several reasons:

- The representation is more compact (*eg* a few dozens coefficients vs. millions of data points) and thus easier to communicate.

- This separates the low frequency (or shape) from the mid/high frequency content. (This is owed to the typical arrangement of polynomial terms from lowest to highest spatial frequency with the increasing index)

2.4.2.5 Simulation of an existing optical system

Some projects driven by optical performance involve running simulations taking into account the shape of parts of the system that were already manufactured, to optimize the rest of the system accordingly. We can also import surface measurements back into our optical design software for straylight simulations or other checks of the as-built system. Given that these surfaces are measured, fixed, the benefit of the link between polynomials and optical aberrations becomes very small and point-based representation can be used. Other considerations such as raytracing speed and representation compacity still apply.

2.4.3 Characteristics of freeform representations

Let us synthesize some characteristics that describe a freeform representation: these are the items to consider when assessing which freeform representation is best for a given context.

- **Availability**: Whether the representation is widely available in software tools or easy to understand and implement. The optical surface definition will likely be transmitted to a few different companies and engineers (design, manufacturing, metrology, simulation checks). If the representation is too esoteric, we run the risk of people misusing formulae (*eg* using explicit formulae for high polynomial orders) or performing lossy conversions.

- **Evaluation speed**: When performing simulations or numerical checks on the surface, the evaluation speed for altitude, slopes and eventually curvatures may matter. In optical design, more systems can be explored in a given amount of time, in straylight simulation more rays can be traced etc.

- **Compacity**: This is the size of the files containing the representation. Polynomials are the most compact, but point-based representations are able to capture the high frequency content in metrology results, at the cost of space. Nowadays storage space is cheap, but file size still matters from a workflow point of view: it makes the difference between exchanging data via email or dedicated file sharing services, or between working on a local machine or on a dedicated server.

- **Approximation error**: The error introduced by switching representations. In the case of a discrete sampling of a polynomial representation to a point-based one, introducing an error is unavoidable. Likewise, in the general case, fitting point-based representations with polynomials or RBF will introduce an error.

Overviews such as this can be helpful to grasp how freeform surfaces are defined, and how each kind has peculiarities which make it more efficient for some applications over other kinds. In our experience, it seems that going into the minutiae of using a polynomial representation only slightly different than another will result in very little benefit. The details for the representations are still useful for implementation and quality control. Usefulness is rather found in switching between broader types of freeform representations depending on the application. For example: 1.polynomial representations for interaction with search algorithms, as the parameters are generally expressive of changes highly linked with optical quality and other performance criteria 2.point-based representations for high-fidelity, high spatial frequency metrology results on an optical surface.

Chapter 3

Freeform Unobscured Telescopes

> Les miroirs feraient bien de réfléchir un peu plus avant de renvoyer les images.
>
> Jean Cocteau - Le Sang d'un poète

Freeform telescopes in the literature – Folding geometries – Different design approaches – Ray-based merit function for telescope design – Aperture sampling schemes – Exit pupil shape – Early straylight prevention – Tolerancing

In the present work, we are particularly interested in the application of freeform surfaces to reflective unobscured imaging telescopes. Freeform optics has brought to instruments greater imaging performance over a larger field of view and aperture as well as greater compacity. The common design approach for freeform telescopes, as implemented in optical design software, involves many details but is largely the same as for conventional optics. One could find room for improvements in each of these details to better accomodate the design of freeform telescopes specifically, which is why we dive deep in the subject.

A lot of design work has already gone into the field of freeform unobscured telescopes, of which we give a brief review. Out of the many folding geometries that give rise to high compacity and imaging performance, we highlight the few that stand out in the literature. We then take a brief look at the variety of design approaches that exist. Digging deeper into the matter, we take a closer look at arguably what is the most popular design approach, which we also adopt, which involves the computation of a ray-based merit function from the discrete sampling of the field and aperture and geometric raytracing. Finally we give a few optical design tips relevant to unobscured telescopes, namely the management of the exit pupil shape and the avoidance of straylight in early design stages. We also touch briefly on the tolerancing of designs.

3.1 Review and Terminology

3.1.1 Examples of reflective unobscured telescope or camera systems

The main differentiating characteristics for reflective unobscured telescopes or cameras are:

- **FOV** (degrees): In the X and Y directions (for rectangular sensors) in object space.
- **Image quality**: Measured either by the Modulation Transfer Function (MTF), Wavefront Error (WFE) or via spot sizes for each field.
- **F#**: F-number of the instrument.
- **Entrance pupil diameter** or **Sensor size**: Missing piece of information to completely determine the characteristics of the instrument. The entrance pupil diameter allows determining the focal length and sensor size from the F# and the FOV, and the sensor size allows determining the focal length from the FOV and the entrance pupil diameter from the F#.
- **Distortion**: Can be seen as local variations of magnification with respect to the field. Does not affect image quality. Can become critical in spectrography instruments.

Muslimov [116] gives a review of the FOV, spot size and F# reached by 17 different TMA designs from the available literature, moreover showing what freeform representations were used for the optical surfaces. The review shows vastly different characteristics across TMA systems, and that no freeform representation is clearly ahead of the others in terms of performance.

We point to a few examples of unobscured telescopes, infinite conjugate camera systems or even HUD from the literature, by number of mirrors, and indicate when the prescription data is given explicitely with the dagger symbol:†

- two-mirrors
  - [115] Two-mirror telescope obtained through differential raytracing.
  - [81] Two-mirror IR imaging system with an accessible aperture stop at the entrance. Prototype manufactured.
  - [117]† Three telescopes with different geometries are given.
  - [118] Two-mirror designs, including reminders about classics such as Schiefspiegler and Yolo.
- three-mirrors
  - [89, 116, 119] Freeform TMAs

  - [12]† Thompson retro-engineered three TMAs with conical mirrors from the NIRSpec instrument on the JWST from data available at the time of writing.
  - [28, 120] Pamplemousse TMA: IR telescope, actually manufactured.
  - [121]† Freeform IR TMA, actually manufactured.
  - [122] Several high performance TMAs are reported that mix different freeform representations.
  - [123]† Two freeform TMAs are given.
  - [124]† Freeform TMA with a linear FOV.
  - [40] Freeform TMA reported.
  - [125] IR freeform TMA, actually manufactured.
  - [126]† TMA with wide push-broom field.
  - [127]† TMA with wide rectangular field, prototype manufactured.
- four-mirrors
  - [128] HUD with four mirrors, including one freeform, one flat and two spherical. The geometry is very compact thanks to folding.
  - [129] Some starting geometries for four-mirrors telescopes are given explicitely.
  - [130]† Four-mirror design strategy and example.

These examples can serve as starting point designs for optical designers. Even when the prescription data is not given, the layout, general characteristics and information about the surface types (spherical, conic, freeform) is often sufficient to generate a system with similar geometry and performance.

3.1.2 Folding geometry

The folding geometry of unobscured telescopes is sometimes used to classify them. It is useful to know the terminology to communicate with other designers.

3.1.2.1 Two-mirror telescopes

Starting from a ray bundle reflecting off the primary mirror onto the secondary mirror, there are only two choices of folding geometry for the secondary mirror. Trumper [117] has coined terms for these, which we adopt here. The two geometries are illustrated on Fig. 3.1.

- **Type 4**: The third ray bundle (3 on Fig. 3.1) is folded back onto the first bundle. The sensor is placed to the left of M1 (when looking from M2 in the direction of the light propagation). The general outline drawn by ray bundles inside the instrument is in the shape of the number 4.
- **Type Z**: The sensor is placed on the right of M1. The general outline is in the shape of the letter Z.

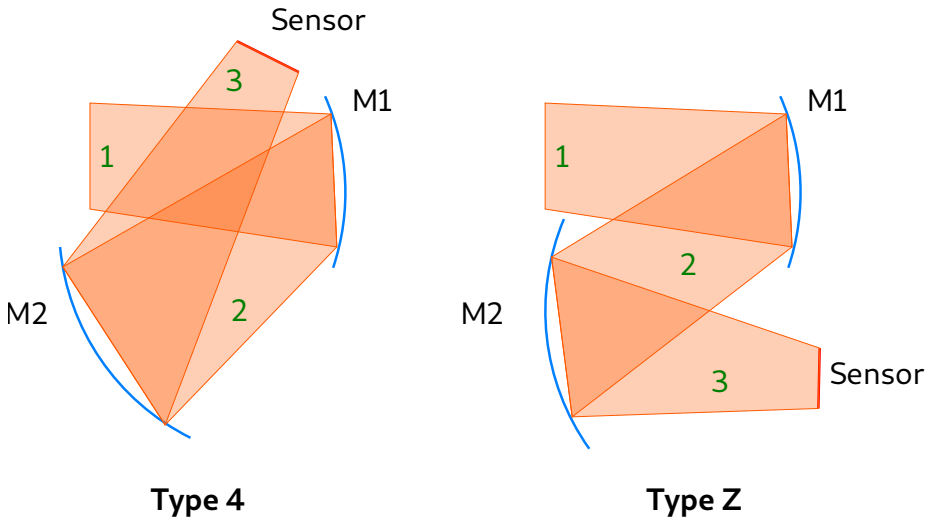


Figure 3.1: The two possible types of two-mirror telescope folding geometry. The ray bundles between diopters are numbered from 1 to 3 from the object to the image.

3.1.2.2 TMA

From the two geometry types for two-mirror telescopes, we can build the possible geometries for TMA, by replacing the sensor with M3 and putting the sensor at various positions between the preceding elements.

Two of the more popular folding geometries are illustrated on Fig. 3.2: the γ "gamma" and W (also *zigzag*) geometries. These geometries give a good compromise between compacity and keeping tilt angles manageable (which is usually good for image quality).

In addition to the folding geometry, the sign (P or N) of the optical power of each of the three mirrors is also used for the classification. A TMA can thus be said to be PNP for concave M1, convex M2 and concave M3. Papa [131] has surveyed exhaustively the combinatorics of folding geometries and optical powers of TMA systems.

3.2 Obtaining Starting Points and Design Methods

We give some references on ways to approach freeform unobscured telescope design (and more broadly freeform optical design). There are several proposed strategies to start designs in the literature, and several families of very different "evaluation approaches" (sequential raytracing to feed a scalar MF into a black-box search algorithm being only one of them).

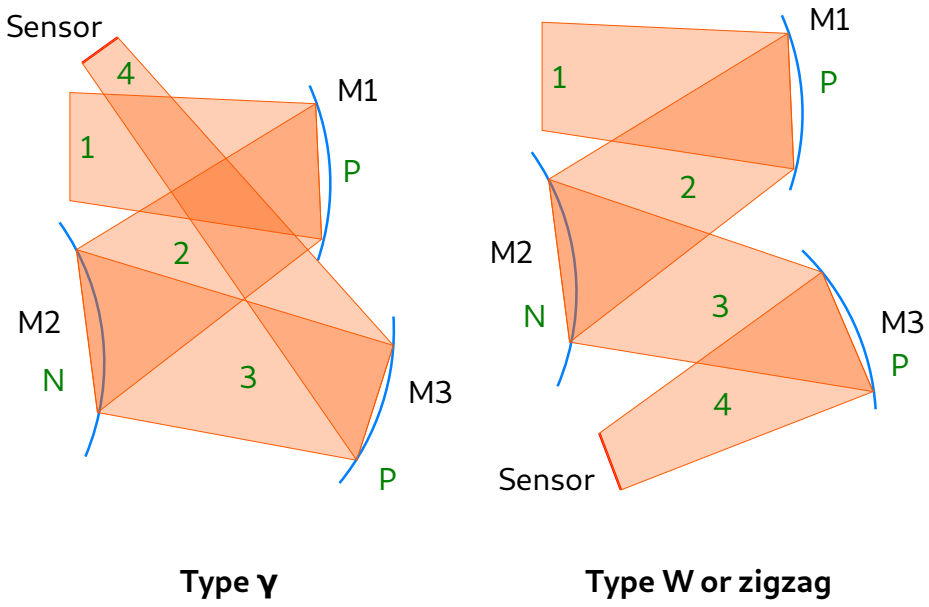


Figure 3.2: Two popular folding geometries for TMA: the γ geometry and the *W*, also *zigzag* geometry. Additionally, these two systems can be said to be of type *PNP*.

3.2.1 Starting a design

In conventional optical design, a typical approach when given a set of instrument specifications is to look for a *starting point* design with similar characteristics from one of these sources:

- **Academic literature**. Papers describing a specific system, its characteristics and performance, and sometimes giving the lens prescription data, are very useful to researchers and engineers.
- **Patents**. The lens prescription data of optical systems can be the basis of a patent [7]. The patented optical systems' performance can be analysed in detail and be used as a baseline for new systems.
- **Collections of designs**. Books such as Smith's [132] offer collections of lens prescription data for systems of all kinds. This is tremendously useful to help dimension a system (number of lenses, achievable performance, *etc*) with respect to a set of specifications.
- **Past experience and private databases**. A veteran optical designer can always draw from past designs. Moreover, companies and research teams specialized in optical design will often curate (or at least they probably should curate) a database of optical systems.

However, there is a problem with freeform optical design: it being newer. All resources for freeform optical design tend to be rarer. While the corpus of

academic papers and doctoral theses is now quite rich, there are fewer (none?) synthetic books on the subject. Patents are also much rarer. Likewise, the experience of optical designers is usually fresher and often found in niche domains: *eg* aerospace, automotive lighting, upstream research projects.

3.2.2 Design strategies

Several strategies have been proposed to go from rough starting points (possibly spherical or with a rotationally symmetric system geometry) to freeform preliminary designs. Let us make a non-exhaustive list:

- [28] (Chapter 5) Fuerschbach shows the process of designing a telescope by starting from an axial, spherical system and adding tilts little by little along with the proper freeform corrections. The corrections are determined by the analysis of aberration fields and NAT.
- [133, 134] These papers propose simple parametrizations of the system geometry for TMAs to arrive at starting systems.
- [121, 126, 127] In addition to a simple geometric parametrization, these works report the success of iteratively growing either the field or aperture, or both at the same time, in designing telescope systems when the geometry has already been found.

These strategies are not mutually exclusive and can all be tried, depending on what leeway there is in the overall system geometry, and on how far into the design process a system is.

3.2.3 Evaluation approaches

The design strategies of which we have just given a brief account actually overlap with the choice of point of view for performance evaluation. A number of broad families of evaluation approaches are used. They are quite different from one another and do not require the same tools, mathematical formalism and know-how.

- **Real ray-based**: This is the most common approach in optical design software. The performance evaluation and analysis of optical systems are done using the raytracing of real rays. It is general, robust, fast and conceptually simple but there is not a lot of information extracted from the systems to guide the designer. This is the favored approach in the present work[1]. Real raytracing is often preferable to pure aberration theory analysis in the case of complex systems (many diopters), with large field of view and large aperture. The reason is that higher order aberration terms for these systems have a significant impact on performance. These higher

[1] This is a bias in the present work and does not represent the state of the literature in freeform optical design. Other methods were not explored as much by the author.

order aberrations are computationally expensive to evaluate and also numerically inaccurate (as increasing exponents are used for field and pupil dependencies)[2]. An example of favoring real ray-tracing over aberration theory for the evaluation of chromaticity in a relatively complex large field and large aperture system is found in [135].

- **Aberration theory**: The analysis of aberration fields and their nodes (NAT) enables a "smart" way of guiding the design, as opposed to the black-box approach of tracing real rays to evaluate a scalar performance MF. Rogers [136] shows in a didactic way how aberration fields can be analysed in unobscured telescopes. Zhong [137] shows how to obtain starting points in TMA systems with analytical evaluation using NAT and Gaussian brackets. Bauer [40] presents a method with NAT to choose TMA geometries that present aberrations readily correctible via the addition of low order freeform caps. Scaduto [138] works out the desensitization to misalignments of a two-mirror telescope through aberration fields analysis.

- **Direct design**: So-called *direct design* methods comprise several related approaches that find discrete sets of points belonging to optimally-shaped optical surfaces, as opposed to polynomial coefficients that describe a continuous surface. [139] lists a good number of these methods in its introduction. [103] shows such a construction method for non-imaging freeform optics. *Direct design* methods come from non-imaging optics but are applied to imagers. The Simultaneous Multiple Surface (SMS) method [140] is another example, applicable both to non-imaging and imaging systems. [113, 124] show construction methods applied to TMA imagers.

- **Differential raytracing**: It involves the computation of Jacobian and Hessian matrices (first and second partial derivatives respectively) for ray quantities (*eg* intersections of rays on surfaces) and merit functions thanks to the differentiability of the optical surfaces representations, the raytracing equations and the merit functions. The derivatives are useful to guide an optimization process or for tolerancing designs. Nice overviews for derivative-assisted system raytracing optimization can be found in [141, 142]. In reverse, optimal freeform surfaces can be constructed via the integration of the derivatives around a known ray [115, 143].

3.3 Ray-based Merit Function

We present general considerations on the computation of merit functions for unobscured telescopes and freeform systems. Compared with conventional optics, we rely less on shortcuts such as quantities computed from marginal rays, which are often invalid in severely off-axis and freeform systems.

[2] Please note however that *eg* TMAs are, as a general rule, not sufficiently complex for high order terms to have an impact that prevents using aberration theory to obtain starting points. Please also note that aberration theory analyses can rely on real raytracing for estimating aberration coefficients. The aberration analysis is then an additional layer of information.

3.3.1 General Definition

In the context of sequential geometrical raytracing simulation of optical systems, and abstracting away the differences between optical design tools, the optical designer will fundamentally use the coordinates of rays inside a given optical system.

Since the search algorithms in most optical design tools work on a single scalar performance metric (which is limiting as we'll see in Chapter 7), called the *merit function*, we have to build it via a combination of individual criteria (*eg* focal length, image quality, *etc*). In order to be independent from particular optical design tools, we give a general purpose MF suitable for most typical freeform telescope design jobs that is purely ray-based: the MF can be constructed only from the geometrical ray data (x, y, z, l, m, n) on the sensor surface[3].

For a given entrance pupil diameter, we need only constrain three criteria for most preliminary optical design of unobscured telescopes, as we have already done in our research paper [144]. We construct three scalar quantities accordingly, for any given field f:

- Spot_f: Image quality through RMS spot radius.
- Pos_f: Image position, *ie* focal length and distortion.
- Tele_f: Image telecentricity.

Quantities related to these criteria are defined on Fig. 3.3. A possible MF (among many others) is given, over all considered fields f, by Eq. 3.1. α and β are arbitrary scalar weights set by the optical designer.

$$\text{MF} = \frac{\sum_{f=1}^{N_{\text{fields}}} \text{Spot}_f + \alpha \cdot \text{Pos}_f + \beta \cdot \text{Tele}_f}{N_{\text{fields}}} \tag{3.1}$$

3.3.2 Image quality: spot size

The image quality can be quantified via a RMS spot radius r_f metric, which is very typical in optical design: Eq. 3.2. This quantity is defined for a given field f and over a given sampling of the aperture stop with N_{rays} rays.

$$\text{Spot}_f = \frac{\sum_{i=1}^{N_{\text{rays}}} (x_i - \overline{x}_f)^2 + (y_i - \overline{y}_f)^2}{N_{\text{rays}}} = (r_f)^2 \tag{3.2}$$

With $(\overline{x}_f, \overline{y}_f)$ the coordinates of the centroid of the spot for field f (Eq. 3.3).

$$\begin{cases} \overline{x}_f = \frac{\sum_{i=1}^{N_{\text{rays}}} x_i}{N_{\text{rays}}} \\ \overline{y}_f = \frac{\sum_{i=1}^{N_{\text{rays}}} y_i}{N_{\text{rays}}} \end{cases} \tag{3.3}$$

[3] Note that what we propose is not new, we give it for the sake of synthesis.

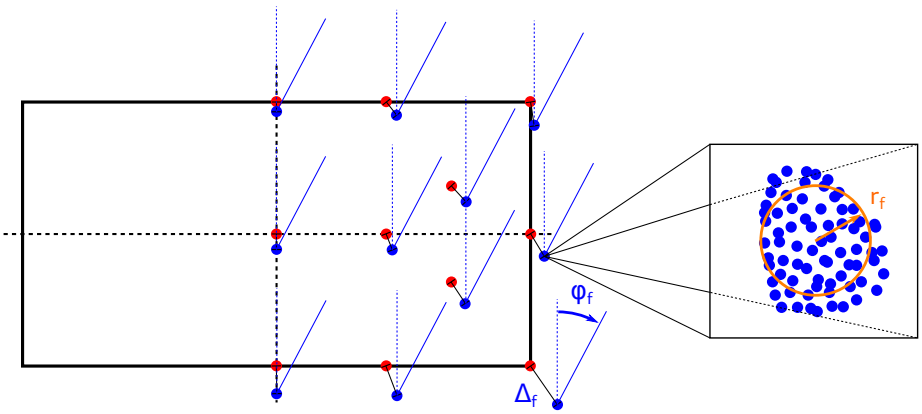


Figure 3.3: Illustration of the three defined MF scalar criteria using image plane ray data. The red dots are targets, or control points. The blue dots are the actual raytraced impacts on the image plane. The dotted blue lines represent the normal to the image plane at the impacts. The blue lines represent the simulated rays incoming on the image plane.

3.3.3 Image position: focal length and distortion

Instrument requirements dictate a mapping between the object field and positions on the sensor. For a given field f, the centroid of the spot must land on target coordinates $(\overline{x}_f^{\text{trgt}}, \overline{y}_f^{\text{trgt}})$ on the sensor. We can construct a quadratic penalty function for spots that wander off from this target[4] (Eq. 3.4).

$$\text{Pos}_f = (\overline{x}_f - \overline{x}_f^{\text{trgt}})^2 + (\overline{y}_f - \overline{y}_f^{\text{trgt}})^2 = \Delta_f^2 \tag{3.4}$$

The question of determining the target coordinates is left to the designer depending on the mission requirements. To illustrate, for instruments with large FOV (such as enabled by freeform optics), we might encounter two types of target mapping from the field to the image:

- Mapping an object plane to the sensor plane homothetically, *ie* a detector pixel is always the same size on the object plane.
- Mapping the object field angles to the sensor plane so that each detector pixel has the same size in solid angle in the object field. This is the case for *all-sky* systems (also for f-theta lenses).

Assuming our instrument has a paraxial focal length f', and noting (θ_x, θ_y) the object field angular coordinates (u-v system) for field f, our target points $(\overline{x}_f^{\text{trgt}}, \overline{y}_f^{\text{trgt}})$ are determined in each case by Eq. 3.5.

[4] One reason to use ray-based metrics for setting the focal length is that the focal length computed in most optical design software (*eg* EFFL in OpticStudio) is a paraxial quantity, which should not be used in large field non-axial systems [68]. Additionally, it is our personal preference to characterize instruments with control points on the image plane and deviations (distortion) from these points rather than simply give the focal length, which can lead to ambiguous interpretations.

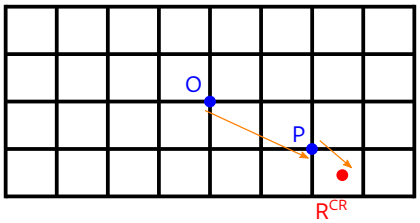


Figure 3.4: Diagram defining some relevant points to the computation of the OpticStudio distortion metric. The plane represented here is the image plane. O (at $(0,0)$) is the center of the field, *ie* the intersection of the chief ray of the center field with the image plane. The black grid is the predicted image chief ray intersections across the defined field range in the hypothesis of no distortion, this corresponds to the *planar* targets (Eq. 3.5). Point P (x_f^p, y_f^p) is a point on that grid for field f. Point R^{CR} (x_f^r, y_f^r) is the real chief ray intersection with the image plane for field f.

$$\begin{cases} \overline{x}_f^{\mathrm{trgt}} = f' \tan(\theta_x); \overline{y}_f^{\mathrm{trgt}} = f' \tan(\theta_y) & \text{planar} \\ \overline{x}_f^{\mathrm{trgt}} = f' \theta_x; \overline{y}_f^{\mathrm{trgt}} = f' \theta_y & \text{f-theta} \end{cases} \tag{3.5}$$

In the *planar* case, a detector pixel is always the same size on a plane object observed perpendicular to the optical axis. In the *f-theta* case, a detector pixel is always the same solid angle in the scene (*eg* in the sky). Please note that what we have just defined is not a metric for *distortion* stricto-sensu, as defined in aberration theory, since we use centroids $(\overline{x}_f, \overline{y}_f)$ rather than chief ray intersections $(x_f^{\mathrm{CR}}, y_f^{\mathrm{CR}})$ and we allow for f-theta targets. Our approach is closer to final instrument requirements[5], but we lose the insights given by aberration theory.

For reference, let us remind ourselves of the definition of the metric for maximum distortion (in percent) that is often given in optical design reports. The usual number is given by the "Grid Distortion" tool in OpticStudio (see the corresponding section in [68]). The relevant points are defined on Fig. 3.4. The zero-distortion grid is estimated according to the usual *planar* targets (Eq. 3.5), an object plane is imaged as an undistorted image plane. The distortion grid is computed using real paraxial rays to approximate the first order properties of the system which are then extrapolated for bigger fields. For any given field f, as per [68], a scalar (percents) distortion metric $D\%_f$ is computed (Eq. 3.6). It corresponds to a signed image plane position error between the real and predicted (with zero distortion) chief ray intersection, normalized by the predicted chief ray distance from the center of the image.

$$D\%_f = 100 \cdot \mathrm{sign}\left(\left[(x_f^r)^2 + (y_f^r)^2\right] - \left[(x_f^p)^2 + (y_f^p)^2\right]\right) \cdot \sqrt{\frac{(x_f^r - x_f^p)^2 + (y_f^r - y_f^p)^2}{(x_f^p)^2 + (y_f^p)^2}} \tag{3.6}$$

[5] In cases where centroids differ significantly from chief ray intersections (*eg* in the presence of coma), what matters ultimately for the mission is where most of the energy is going, as measured by the centroid position

What is usually reported in optical design reports is the maximum absolute value of $D\%_f$ across all fields. Depending on the instrument specifications, it can be useful to abuse the metric definition and use a f-theta field or centroids instead of chief rays.

3.3.4 Image telecentricity

Image telecentricity can be defined as a condition met when the exit pupil is at infinity in the image space. Alternatively, the ray-based approach is that all chief rays across the field are parallel to one another. Telecentricity in the image space can be critical when plates with surface treatments or filters are involved, their response being a function of the angle of incidence. It is also needed to minimize crosstalk between pixel channels.

We can build a metric penalizing the difference between the angle of incidence ϕ_f of different chief rays across the field sampling (Eq. 3.7). The metric is zero when all chief rays have the same angle of incidence on the image plane.

$$\text{Tele}_f = (\phi_{x,f} - \overline{\phi}_x)^2 + (\phi_{y,f} - \overline{\phi}_y)^2 \tag{3.7}$$

With $\overline{\phi}_x$ the angle of incidence of chief rays on the image plane averaged over the field samples (Eq. 3.8).

$$\overline{\phi}_x = \frac{\sum_{f=1}^{N_{\text{fields}}} \phi_{x,f}}{N_{\text{fields}}} \text{ , idem for } y \tag{3.8}$$

3.4 Sampling the field and the aperture stop

As with most optical system, we derive our ray-based metrics from a sampling of both field and aperture spaces. The number of samples is the result of a compromise between:

- **Metric accuracy**: The computed metrics for distortion, image quality and telecentricity must describe the actual system performance with sufficient accuracy for the purpose of optimization. *Eg* a field sampling with rays from just the center field and one extreme field will likely be insufficient to describe the performance accurately, we must also sample the fields in-between and in other directions.
- **Computation time**: The computation time for the merit function grows linearly with the number of rays. Sampling the aperture or field in too fine a fashion will result in a lower number of optical systems being evaluated per unit of time in search algorithms.

3.4.1 Pupil

For a given field, rays sample the wavefront in pupil space. The metrics that are integrations on the pupil (such as rms spot radius) are only approximations when a finite number of rays are used.

Let us look, as an example, at the exact definition for the RMS spot radius r_f, as explained by Malacara [145] (Section 8.4.1). Taking the limit of an infinite number of rays sampling the pupil space in uniform fashion, on a well-behaved optical system (the aperture stop is taken to be a continuous disk), the set of ray impacts on the image plane is a continuous function of the position (ρ, θ) on the pupil disk into $\mathbb{R}^2$, (x, y) on the image plane, $(\rho, \theta) \mapsto (x, y)$. Additionally, this function is dependent on the field f considered (also in a continuous fashion for most optical systems). Using this continuous point of view, r_f becomes a standard deviation with euclidian distance on the image plane, integrated over the pupil:

$$(r_f)^2 = \frac{1}{\pi} \int_{-\pi}^{\pi} \int_{0}^{1} \left[(x(\rho, \theta) - \overline{x}_f)^2 + (y(\rho, \theta) - \overline{y}_f)^2 \right] \rho d\rho d\theta \tag{3.9}$$

The centroid $(\overline{x}_f, \overline{y}_f)$ also assumes an integral definition of course. Eq. 3.9 is found as eq.8.36 in [145], we removed all assumptions of symmetry however.

The goal of pupil sampling schemes is then to approximate the "pupil-integral" quantities, such as spot radius, using as few rays (samples) as possible. These sampling schemes are already well integrated in optical design software and in the literature [146, 147]. The same problem (or at least very related) appears in surface wavefront fitting in metrology. This is also a common problem in integration via monte carlo methods, there is a large applied mathematics literature treating this *cubature* problem[6]. We can give a few usual sampling schemes in a non-exhaustive list, grouping broad types together:

- **Regular grids**:
  - **Cartesian**: The coordinates (x, y) are sampled at regular interval, and cropped by the aperture disk.
  - **Polar**: Sampling at regular interval the polar coordinates (ρ, θ). Suffers from a lack of points near the edge.
  - **Polar with equi-arcs** (fig.2 in [146]): The polar plane is sampled at regular ρ interval, but the interval in θ is computed such that points are always spaced by the same arc length across all rings. This results in a sampling more uniform than its naive polar counterpart.
  - **Hexagonal** (fig.5a in [76]) Honeycomb pattern.
- **Spirals**: Can be expressed as a polar parametric equation $t \mapsto (\rho, \theta)$ and sampled at regular t intervals.
  - **Logarithmic** [148] $\rho(\theta) = ab^\theta$ with $a > 0$, $b > 0$.

[6] *cubature* names the numerical integration of higher dimensional (more than 1) functions.

  - **Archimedean** [149] $\rho(\theta) = a\theta^{\frac{1}{n}}$[7].
  - **Fibonacci** (special case of logarithmic) [150–152] (see Fig. 3.5)

- **Low-discrepancy sequences**: Deterministic sequences of points that have the nice property that points do not cluster too much yet sample space in uniform fashion (this is, roughly speaking, what low discrepancy is) and in a "random-looking" way. These are compromises between random samplings (the points of which tend to form undesirable clusters) and a regular patterns (points are spread out but we may generate frequency problems in the sampling due to the regularity) [153].
  - **Halton** (see Fig. 3.8)
  - **Sobol**, **Hammersley** *etc.*

- **Stochastic**: These are random sampling methods.
  - **Random uniform** [154] (see Fig. 3.7). It suffers from naturally forming clusters of points. For most applications, it is useless to sample locally with a very high density.
  - **Poisson-disk** [155]: Random sampling method with the additional rule that no two points should be closer than some disk radius.
  - **Edge-clustered** [76]: Random sampling method proposed by Kaya *et al* for the approximation of freeform optical surfaces. It is interesting to sample the edge of the disk more densely since this is where large altitude variations for most polynomials occur (see *eg* Zernike). In the case of aperture sampling, the rays near the edge will also likely be those that contribute the most to degrading the image quality metrics, and thus need to be included in the sampling more so than well-behaved paraxial rays.

- **Quadrature rules** We mention here weighted schemes for sampling. Points are assigned a weight and a position according to some rule constructed to allow the approximation of the integral of families of functions. Forbes has shown decades ago that these could be used successfully for computing spot radii [146] and it seems to still be in use today.
  - **Radau**
  - **Gaussian**: used *eg* in OpticalStudio (see [68] Gaussian quadrature) (Fig. 3.6).

[7] The spirals with negative n are asymptotically collapsing towards their center. This does not seem a very useful property for sampling the unit disk.

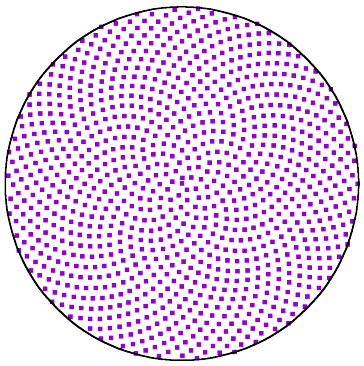


Figure 3.5: Sampling on a Fibonacci spiral. 1000 points.

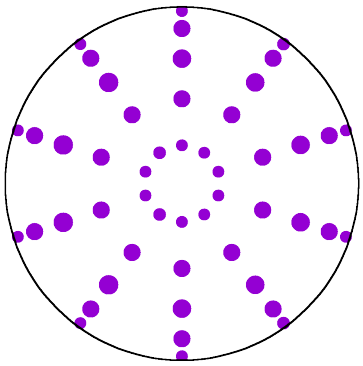


Figure 3.6: Gaussian quadrature sampling with 5 radial steps and 10 angular steps. The area of each circle is proportional to the weight. The positions and weights are taken from Forbes [146].

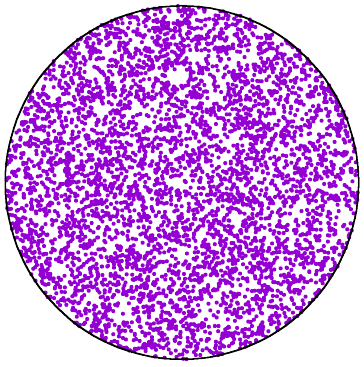


Figure 3.7: Random uniform sampling. Notice the small clusters of points that form naturally. Sampling points that are too close to one another is generally useless.

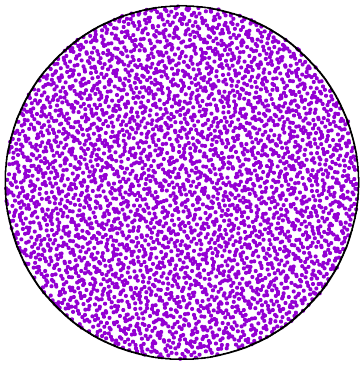


Figure 3.8: Halton sequence with bases 2 and 3 for the x and y coordinates respectively. Notice that in comparison to random sampling, the clustering effect is reduced and the disk is still sampled in a uniform fashion.

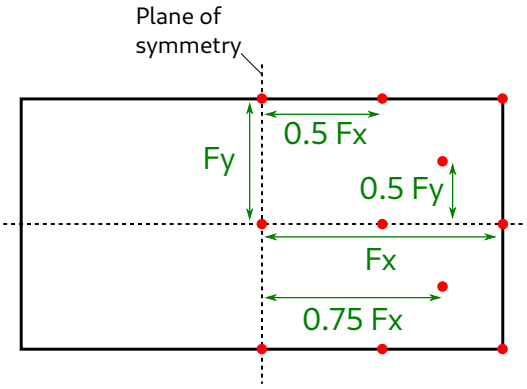


Figure 3.9: A general purpose field sampling for plane-symmetrical instruments with a rectangular field.

3.4.2 Field

Since most unobscured reflective telescopes possess a plane of symmetry, there are some redundant rays. We could very well choose to trace only half the aperture stop sampling and both halves of the field in a symmetric fashion. However, the usual choice is to sample the aperture stop completely and symmetrically and only half of the field. One possible general purpose field sampling is showed in Fig. 3.9[8]. A similar but denser sampling is reported by Trumper *et al* [117].

3.5 Pupil Shape Control

A common issue with off-axis systems is that their exit pupil will often be non-circular. One of the consequences is a non-circular Airy pattern, which complicates image analysis and degrades the MTF in the elongated direction. Expressed in another way, this means the MTF diffraction limit depends on the direction in the object field. For a given field and in the case of near image telecentricity, the exit pupil shape can be assessed in a virtual plane perpendicular to the chief ray in the image space (Fig. 3.10). An example of marginal rays intersecting such a virtual plane for a real freeform system is given in Fig. 3.11.

There is a simple geometrical criterion we can use to constrain the exit pupil circularity. In the image space virtual plane perpendicular to the chief ray, we trace N_{marg} marginal rays. From this ray data, we can compute a metric penalizing the non-circularity of the shape drawn by the rays (Fig. 3.12) for a given field.

[8]The sampling is here limited to 11 points because the number of fields is limited to 12 in Zemax 13. In any case, more fields can be used in the MF when it is user-defined. In the latter case, we are limited only by the computational expense.

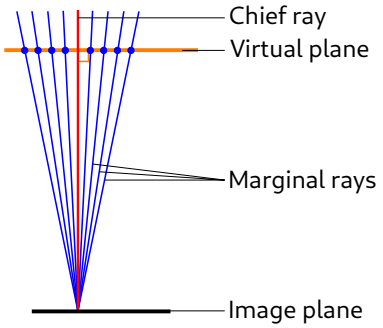


Figure 3.10: The shape of the exit pupil can be assessed from a set of marginal rays in a virtual plane perpendicular to the chief ray in the image space.

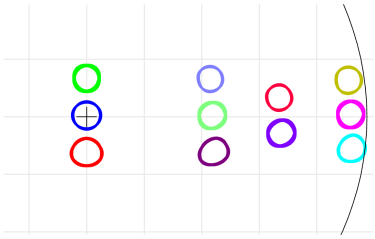


Figure 3.11: Example of the intersection of marginal rays on a virtual plane away from the image plane, as traced by OpticStudio. The different colors represent different fields. Note some shapes deviate slightly from circles, both due to image aberrations and pupil shape aberrations.

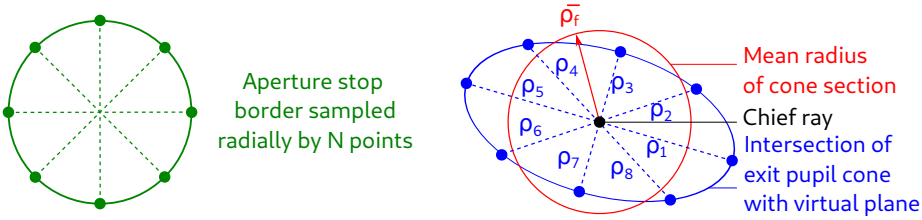


Figure 3.12: Diagram explaining one way to constrain the circularity of the marginal rays on a virtual plane in image space.

$$\text{Pupcirc}_f = \sum_{i=0}^{N_{\text{marg}}} (\rho_i - \overline{\rho}_f)^2 \tag{3.10}$$

Note that this is an empirical metric useful in the case of roughly telecentric systems. We have not investigated system with large angles of incidence on the image plane. We also have not investigated the illumination inside the pupil, from the entrance pupil to the aperture stop and then from the aperture stop to the exit pupil.

3.6 Straylight

Straylight analysis is an important part of the design of any instrument. Straylight can limit the Signal-to-Noise Ratio (SNR) of an imaging system, to the point of, in severe cases, completely blinding it. A full-fledged straylight analysis requires that every part of the instrument be known: the optical surfaces of course, but also all the mechanical parts, the windows and coatings, the baffling,

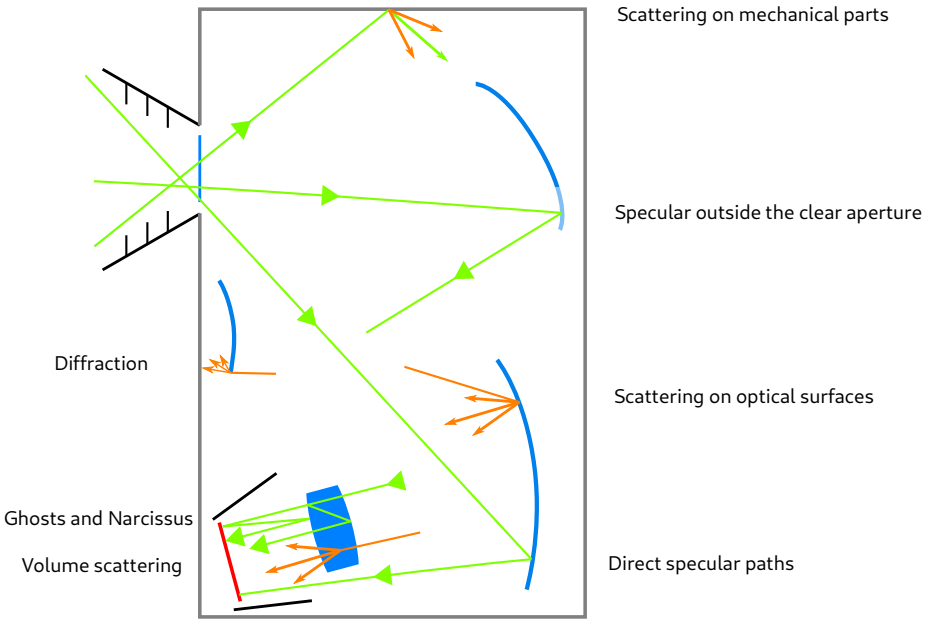


Figure 3.13: Diagram of the different types of straylight inside a telescope.

and a good knowledge of the Bidirectional Reflectance Distribution Function (BRDF) for all the materials found in the instrument. Special attention must be given to straylight in the design of unobscured telescopes since compacity increases the risk of significant straylight occuring.

We illustrate the different possible sources of straylight on Fig. 3.13. In order of decreasing typical quantity of generated straylight levels, we can list:

- *Direct specular paths*: These are the straylight paths that follow a non-nominal specular path from the instrument entrance window to the sensor. The straylight level is of the same magnitude as the observed object.
- *Specular paths outside the clear aperture of optical surfaces*: A potential snag for designers is the straylight coming from the outer part of optical surfaces. In the early stage of optical design, the raytracing usually includes only the useful aperture of the optics (cut as close as possible to the ray footprints). The optical parts will then be usually manufactured with slightly larger clear apertures and straylight can be generated by the added optical areas.
- *Reflections off the mechanical parts*: Either specular or diffusive reflections happen on the mechanical parts. These are modelled by the BRDF of the material. In worst case scenari, with highly specular materials, their power can be a significant portion of the useful flux.

- *Ghosts and Narcissus*: Ghosts can be generated either by the Narcissus effect, *ie* light reflecting off the sensor back into the optical system and then back to the sensor again, or by refractive optics. The typical maximum relative straylight level of these stray paths is in the order of magnitude of 1%.
- *Surface scattering*: In addition to the nominal specular reflection, all optical surfaces have surface defects of various sizes that generate a small amount of diffused light.
- *Volume scattering*: For refractive optics, light is diffused not only by the diopters but also in the volume itself.
- *Diffraction*: Typically on the edge of the aperture stop, some light is diffracted in (generally) unintended directions.

To avoid the need for heavy redesigns, it is advisable to take into account the straylight early in the design process. As it happens, even during preliminary optical design, we already have enough data on the system to avoid most of the major causes of straylight: the direct specular paths, the specular paths generated by the outer aperture of optics and even ghosts.

We give some diagrams of how the design of unobscured telescopes avoiding direct specular straylight can be approached, in a very practical and visual way, by the optical designer. First, we give a diagram in Fig. 3.14 (on a hypothetical "zigzag" 4MA) highlighting the envelopes of rays between diopters. It is useful to think in terms of envelopes to see what spaces are potentially free to place optical baffles. Using the intersection between successive envelopes, we can deduce the envelope of the straylight rays that will reach a particular surface no matter the baffling. Using this, we can determine what part of the optical surface is exposed to straylight, noted S on Fig. 3.14. The optical designer can then evaluate visually the severity of the degradation this direct straylight can cause.

We give in Fig. 3.15 a simple visual criterion to detect whether a straylight ray is likely to reach the instrument's sensor through a direct specular path. On a given mirror (M3 in this example) that is not in a pupil plane, we note C the cone of all possible incident angles for nominal imaging rays, irrespective of the intersection's position on the mirror. Assuming a near-spherical surface, any ray that hits M3 with an incident angle within this cone will likely land somewhere in the nominal image plane of M3, here the instrument's sensor. We see on Fig. 3.15 an example of a straylight ray that contaminates the image plane of M3 in such a way. For the particular example of the straylight ray drawn in Fig. 3.15, it is as if the ray were coming from within the nominal field of the instrument but from a slightly larger aperture stop, this is why it is highly likely that it will contaminate the image.

We have omitted external baffling in this section, but it is a common type of baffling to block straylight coming at angles outside the nominal field. It is commonly cone-shaped and placed around the entrance window. The demerit

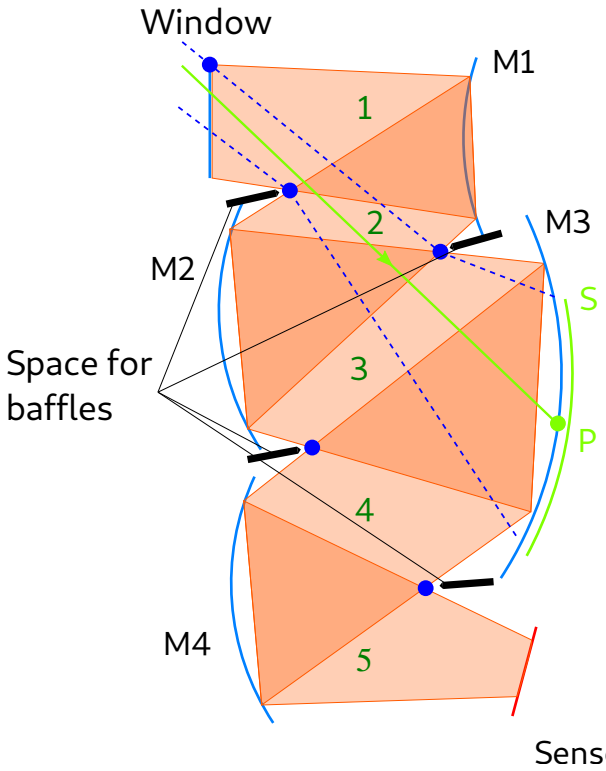


Figure 3.14: Diagram illustrating a way to approach straylight analysis at an early design stage. The optical surfaces of the instrument are in light blue. The orange parallelograms numbered from 1 to 5 are the envelope of all nominal rays between each pair of successive diopters. Obstructing these envelopes means that an obscuration will occur. The spaces left free between the envelopes are potential locations for optical baffles. The intersection (here in 2D) between two successive ray envelopes is drawn as a dark blue dot. The dotted blue lines represent the envelope of all straylight rays that pass through the window towards M3 and cannot be baffled without creating an obscuration. The set of all "un-bafflable" straylight rays from the window to M3 has the footprint S on M3. A particular straylight ray is drawn in light green and intersects with M3 at point P.

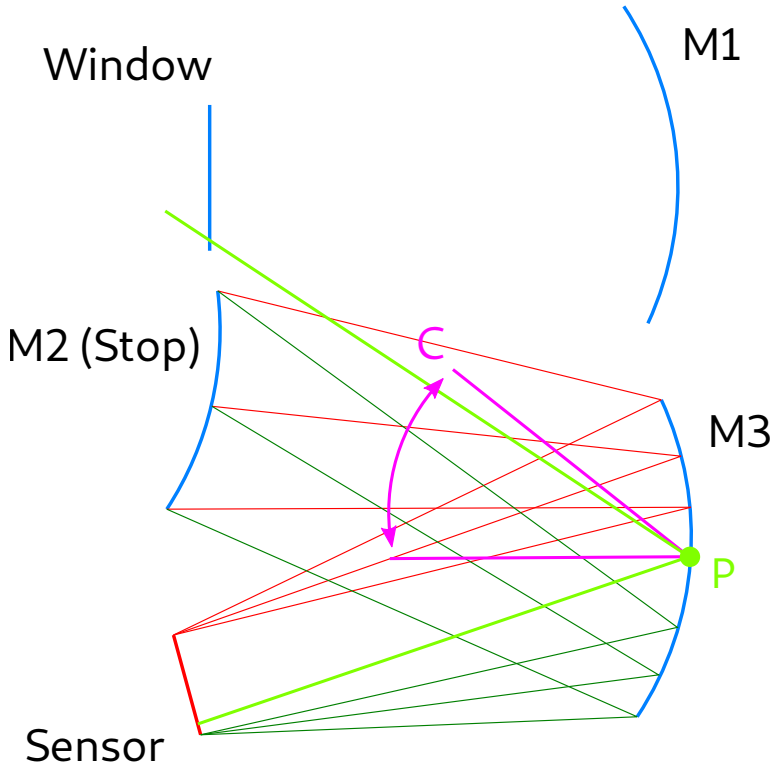


Figure 3.15: Diagram illustrating a visual method for assessing whether a straylight ray will follow the nominal specular path of the instrument, thus having a high likelihood of reaching the sensor. We assume near-spherical optical surfaces. A straylight ray in light green passes through the window and intersects M3 at P. C is the cone of all possible incident angles for nominal imaging rays on M3. When a straylight ray is incident on M3 with an angle within the cone C, then it is likely that it will reach the sensor.

of this type of baffling is that it can potentially be quite long, as designers are tempted to make it longer to block rays at incident angles closer and closer to the nominal field.

It is understood that the method we give is vague and mainly a visual aid for optical designers. Developing a rigorous criterion robust across all instrument geometries could prove difficult but worthwhile. A more traditional straylight analysis of all specular paths (as shown in Chapter 4) remains the surest way to quantify straylight in an instrument. It is however often impractical in the current state of optical design software to go back and forth between sequential raytracing and straylight analysis, which is why such visual aids as we have given are useful in the earliest stage of design.

Also note that all the diagrams we have drawn are in 2D for convenience. In reality the envelopes of rays are in 3D, as well as the baffling etc. From experience, we know straylight can take unexpected paths through an instrument that are not easy to visualize on 2D cross-sections and must be displayed in a 3D view.

We summarize the types of specular straylight the optical designer must pay attention to in early design stages, in order of increasing severity:

- Straylight that enters the instrument but does not reach a specular surface. This type is mostly unavoidable and will be quantified and mitigated (if necessary) much later in the design process.
- Straylight that enters the instrument and reaches a specular surface, but never reaches the sensor through a specular path. It is largely unavoidable but better if minimized since surface scattering will generate straylight that cannot be mitigated through baffling.
- Straylight that reaches the sensor through a direct specular path but can be blocked by eventual baffles. Adding optical baffles everywhere in the instrument is not a viable strategy since other mechanical parts must fit, it is better to minimize this type of straylight through the design geometry.
- Straylight that reaches the sensor through a direct specular path and cannot be blocked by baffles (as seems to be the case in Fig. 3.15). There is no way to mitigate this type of straylight other than through obscuration, a change of design or sometimes very long external baffling. It must be avoided since there is potentially no way to fix the problem later in the design process and the amount of straylight is likely very significant.

3.7 Tolerancing

Tolerancing in optical design is the study of the influence of perturbations to the nominal design on the system performance. In much the same way as straylight analysis, tolerancing is better done as soon as possible in the life of a design to detect dead-ends but requires data from subsequent steps in order to become accurate. The task of the optical designer is often not to find the best nominal

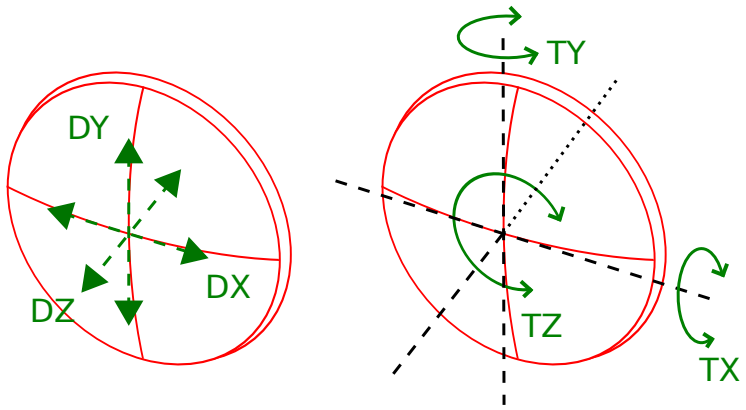


Figure 3.16: Diagram illustrating 6 DOF that can affect a mirror on its mount. Reproduced from [158].

design but the design that will be resilient to the largest errors while maintaining acceptable performance. We usually distinguish between two kinds of perturbations:

- **Position**: The error in position and orientation that affects diopters due to manufacturing errors or alignment errors. We usually consider all 6 DOF of rigid-body transformations (see Fig. 3.16).
- **Shape**: Optical surfaces are never manufactured exactly to specifications. We usually take into account low spatial frequency deviations first. Rugosity (highest frequency) belongs to straylight analysis. Mid-spatial frequencies, although important, are difficult to model [156, 157].

3.7.1 Position

Tolerancing the position of elements is more straightforward than tolerancing shape defects. One possible method is to draw perturbations in all 6 DOF according to a gaussian distribution centered at the nominal position. We manually tune the standard-deviations to some realistic values, starting from large errors and progressively reducing the error budget. We evaluate a sufficient number of systems to obtain statistically significant results. We consider a design to be resilient to the applied perturbations when a sufficient proportion P_{tol} of systems (say 90%) still perform above the target requirements. The result of this analysis is a set of standard-deviations that are as large as possible so that performance is still acceptable for a proportion P_{tol} of randomly drawn systems. The usefulness of this analysis can be improved by knowing in advance the type of mounts that will be used and the usual errors associated with them, *eg* what standard-deviation can we expect for each axis of translation and rotation from our experience on previous systems. Please note DOFs can be broken down,

depending on the system idiosyncrasies, into sub-DOFs relating to centering errors in manufacturing, mount and assembly error etc. An example of a relatively detailed tolerancing for the MAORY instrument is found in [159]. An analysis for Pamplemousse is found in [28] (section 7.1.1).

3.7.2 Shape

Tolerancing surface shape errors is challenging. We usually indicate an error budget in RMS altitude deviation from the nominal shape. The way in which the altitude deviation should be randomly drawn is however not clear at all. Let us take two extreme examples to understand why. An optical designer could be tempted to use:

- The first low-order terms of a freeform polynomial representation. But then, why choose a particular polynomial basis rather than another one? Why stop at a fixed number of terms, don't higher frequencies also occur in manufacturing? How would we choose the range in which to randomly draw the freeform coefficients, is there any reason why the range would be identical across coefficients?
- A grid of points. If the altitude of individual points are drawn randomly in an uncorrelated fashion, we are left with nothing but rugosity. If we choose to draw gaussian bumps, how would we calibrate their amplitude, what about their lateral size? Why would the bumps be gaussian?

In an ideal world, the optical designer would know how each diopter would be manufactured and have statistical data on the type of surface shape errors that appear for the given manufacturing machine, for this given shape etc. Obviously, this is almost never the case, except perhaps for optical surfaces produced for mass markets. The optical designer is then left to find an acceptable compromise to somehow model random surface errors in a credible way. The experience from past projects (*eg* measurements of previously manufactured optics) is precious.

The strategy that we have applied for the design of a TMA [158] (another strategy is found in [159]) is to use Zernike Standard coefficients. Since the individual coefficients represent directly a RMS altitude contribution, we make the assumption (still quite arbitrary) that we can treat every coefficients the same way. Our strategy is[9]:

1. Determine a fixed value for RMS altitude error.
2. Choose Zernike standard terms to include in the shape tolerancing. Piston, tip and tilt are excluded since they are already in the position tolerancing analysing.

[9] This method can be readily implemented in OpticStudio with TEZI operands, albeit with a few scripting tricks.

3. Distribute the fixed RMS budget at random uniformly across the chosen Zernike coefficients[10].

Please note this method comes with a few caveats and is not entirely satisfactory:

- The choice of the included Zernike terms can make the results vary quite a bit. For example, selecting all terms between Z4 and Z45, rather than just Z4 through Z28, will have the effect of reducing the expected error amplitude of all low-frequency terms, the rms contributions will be spread over a larger frequency spectrum. It might be worthwhile to specify error budgets using pre-defined ranges of Zernike terms, as in [159].
- There is no clear reason why the Zernike polynomial basis would be the most adequate to describe manufacturing errors arising from across all machines and methods.

3.8 Useful tools relating to polynomials

In the context of optical design with freeform optics, we often want to:

- Constrain the RMS or maximum departure in altitude or slope. We usually want to avoid departures that are larger than necessary. To this end, we can craft penalization terms, either using a general method that samples the optical surface (general but computationally costly) or with exact methods when the orthogonality property of the polynomial basis allows it (*eg* Zernike for altitude, Q-polys for slope) (see Section 2.1.1.6). See also Takaki *et al* [160] on applying departure regularization terms in the context of TMA design.
- Convert a surface between different polynomial basis. Manufacturers will often ask for a specific freeform basis (usually Zernike). It can also be used for the purpose of optimization: converting a surface from one representation to another necessarily changes the merit function landscape and can be used advantageously to get out of local minima.[11]

As we have seen, optical designers have different approaches available for freeform telescope design. Firstly the choice of the freeform representation, most likely polynomial and as shown by the literature, is not likely to be a huge influence on the final design performance. Switching between polynomial bases

[10] This can be done by drawing a random uniform number for each coefficient between 0 and 1, and then scaling all the coefficients so that the total RMS contribution matches the defined RMS error.

[11] As already mentioned in Section 2.1.3.1, Forbes cites a method for efficient change of basis in [90]

can be advantageous for reasons of commodity, perhaps modify the relation between parameters and MF when the system is "stuck" in the case of the use of search algorithms (as will be explained later), or taking advantage of orthogonality properties to access quickly-computed slope or height limiting constraints. Understanding the details of ray-based optical simulation is sometimes required when very specific design specifications must be met and custom computations performed (either through commercial software existing scripting capabilities or through external software). We insist here on the value of taking into account straylight, tolerancing and pupil shapes, which can generate a slew of problems further down the road and are better dealt with as early as possible. The following chapter will showcase the design of a TMA system in application of the design approach described in the present chapter.

Chapter 4

YATMA: TMA Design Report

General characteristics – Nominal performance – Tolerancing the position and shape of mirrors – Analysing straylight – Prescription data – Discussion of pros and cons of this design

We present the design of a compact freeform TMA telescope suitable for thermal IR applications with rectangular field of view, near diffraction-limited performance and large aperture. We nicknamed the design YATMA, for Yet Another TMA. This design is a rework of the one we have presented in previous work [158], and was part of the CNES R&T FAME supervised by Vincent Costes. We give an overview of the system and its performance, and provide all the data required to replicate the design.

4.1 Overall characteristics

This TMA is the result of a study for a thermal infrared telescope for nanosatellites. The high level specifications, including focal length and minimum aperture were provided by CNES. The volume constraint, as well as the aperture were critical and explain the recourse to freeform optics. An all-ceramic mirror blank was manufactured using additive manufacturing technologies to assess the feasibility of this technique.

We give the general characteristics of YATMA in Tab. 4.1, the system layout in Fig. 4.1 and a CAD view of the optical components in Fig. 4.2. The sensor specifications are taken from Lynred's "PICO1024 Gen2" sensor.

Classification-wise, our TMA is a PNP W (or zigzag, see Section 3.1.2.2) with the aperture stop on M2. All three mirrors are freeform, using plane-symmetric Zernike Standard terms up to Z37 (see Appendix A).

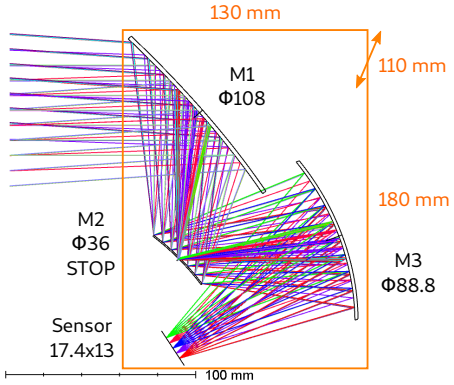


Figure 4.1: YATMA layout and physical dimensions.

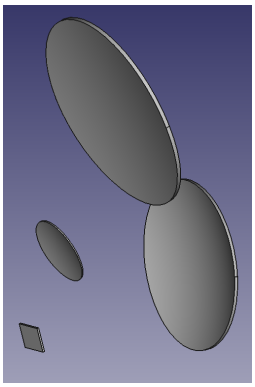


Figure 4.2: CAD view (Freecad) of YATMA (optics only).

F-number	1.84
Airy disk diameter @8 µm	35.9 µm
Full field (rectangular)	$10.7° \times 8°$
Volume	$130 \times 180 \times 110$ mm or 2.6 L
Max. distortion (see Eq. 3.6)	3-4 %
Min. MTF @30 mm^{-1}	0.39
Sensor: Uncooled microbolometers	
Pixels	1024×768
Pixel pitch	17 µm
Sensor area	17.4×13.1 mm
Spectral band	8 - 14 µm
Nyquist frequency	30 mm^{-1}

Table 4.1: General TMA characteristics.

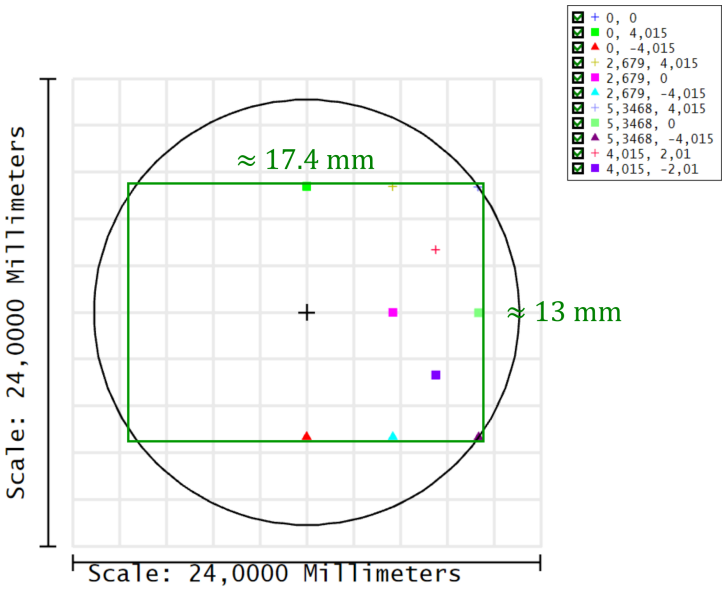


Figure 4.3: Field sampling used for the optimization and performance analysis. The system symmetry is exploited here. The values in the legend are object field angles in X and Y directions in degrees. The diagram represents the nominal position of these fields on the sensor plane (in the absence of distortion).

4.2 Performance analysis

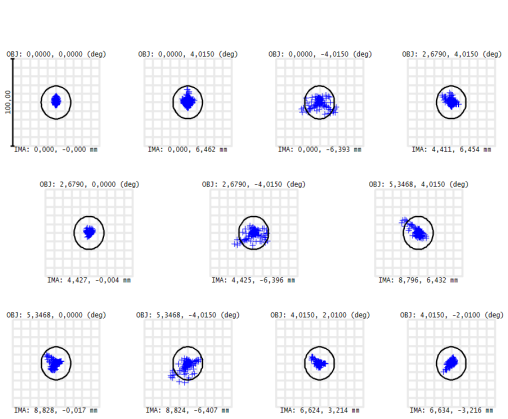
We analyse the performance of YATMA using typical metrics for optical design. The field is sampled according to the scheme illustrated in Fig. 4.3. The spot diagrams for these fields are in Fig. 4.4. So-called *diffraction* and *Huygens* MTF are plotted on Fig. 4.7 and Fig. 4.8. Both the spot diagrams and the MTF show that the performance is *near* diffraction-limited. The exit pupil being slightly elliptic, we see two bundles of MTF curves depending on the direction of analysis. In addition, we check that there is no performance drop-off hidden in the field by plotting the RMS spot radius (Fig. 4.5) and the RMS wavefront error (Fig. 4.6) on a finely-sampled field grid.

4.3 Tolerancing and Straylight Analysis

4.3.1 Tolerancing

The tolerance study involves introducing the following errors:

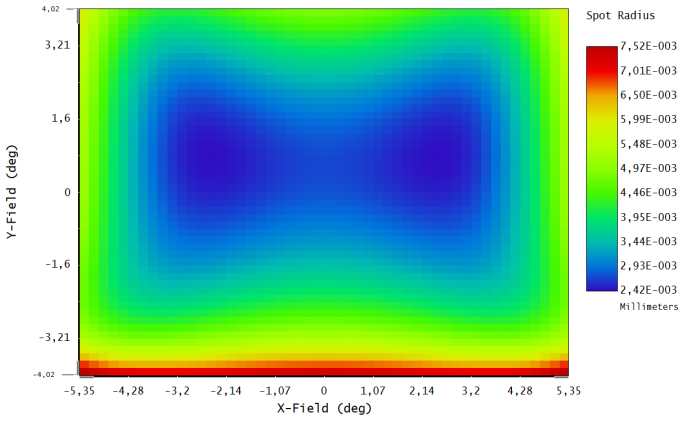
- Position: rigid body DOFs of the mirrors and sensor. Perturbations are drawn along each axis or angle according to a normal distribution of a given standard deviation.



Surface: IMA

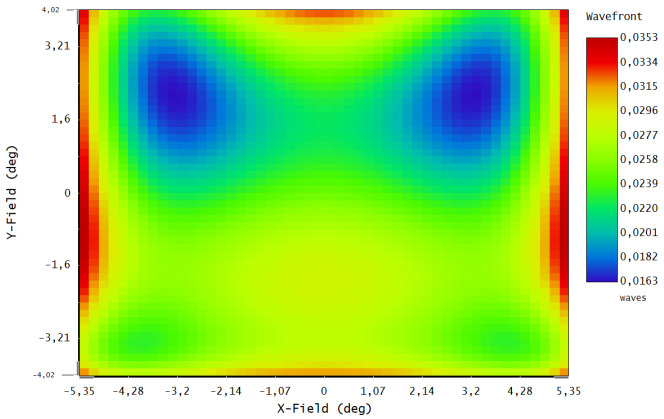
Spot Diagram	
19/09/2018 Units are µm. Airy Radius: 17,94 µm. Legend items refer to wavelengths Field : 1 2 3 4 5 6 7 8 9 10 11 RMS radius : 3,018 5,126 8,618 4,856 2,723 8,134 6,706 5,463 8,590 3,648 3,872 GEO radius : 7,559 14,907 22,855 13,395 6,032 25,529 23,012 14,746 25,092 11,706 13,184 Scale bar : 100 Reference : Chief Ray	Zemax Zemax OpticStudio 18.4.1 TMA_v2.1_nom.ZMX Configuration 1 of 1

Figure 4.4: Spot diagram for considered field sampling. The black circles are the Airy disks @$8\,\mu\mathrm{m}$.



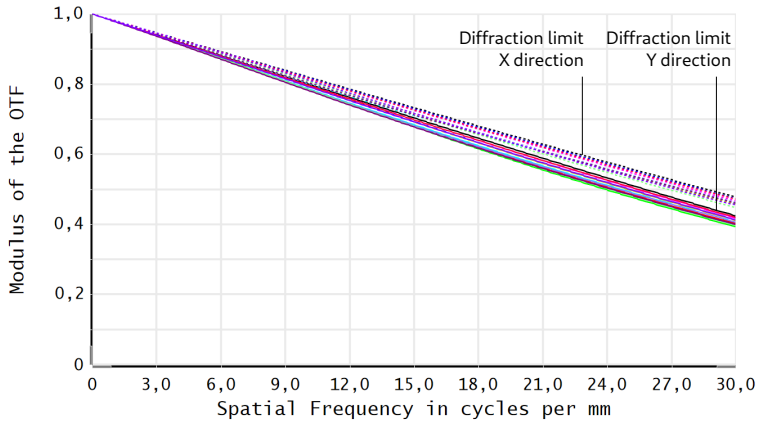
RMS Spot Radius Field Map	
19/09/2018 Field Size X = 5,3468, Y = 4,0150 degrees Min RMS = 0,0024, Max RMS = 0,0075 Millimeters Wavelength: Polychromatic Surface: Image	Zemax Zemax OpticStudio 18.4.1 TMA_v2.1_nom.ZMX Configuration 1 of 1

Figure 4.5: RMS spot radius across a fine sampling of the field.



RMS Wavefront Field Map	
19/09/2018 Field Size X = 5,3468, Y = 4,0150 degrees Min RMS = 0,0163, Max RMS = 0,0353 waves Wavelength: Polychromatic Surface: Image	Zemax Zemax OpticStudio 18.4.1 TMA_v2.1_nom.ZMX Configuration 1 of 1

Figure 4.6: RMS wavefront error across a fine sampling of the field. The wavelength is $8\,\mu\mathrm{m}$.



Diffraction MTF	
19/09/2018 Data for 8,0000 µm. Surface: Image Legend items refer to Field positions	Zemax Zemax OpticStudio 18.4.1 TMA_v2.1_nom.ZMX Configuration 1 of 1

Figure 4.7: Diffraction MTF for considered field sampling @$8\,\mu\mathrm{m}$. We stop at the sensor's Nyquist frequency. The topmost black curve in each of the two bundles is the diffraction limit. The X direction on the sensor is perpendicular to the plane of symmetry of the system.

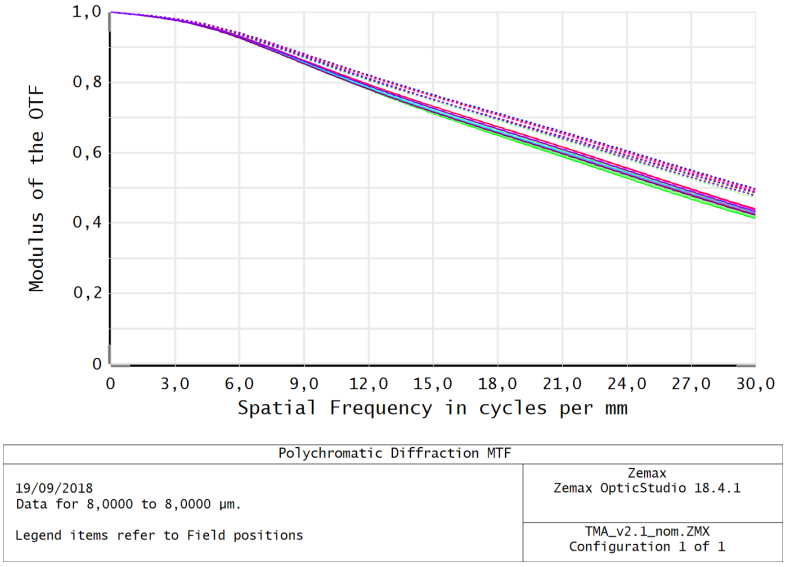


Figure 4.8: Huygens MTF for considered field sampling @8 µm.

- Surface shape: shape error on the mirrors. The errors are drawn using Zernike Standard coefficients Z4 through Z45 using the strategy explained in section Section 3.7.2 with fixed RMS contribution spread randomly over coefficients in the chosen range.

We report in Tab. 4.2 the standard deviations for the gaussian distributions that were used so that 90 % of systems reach an average (across considered fields) diffraction MTF value at Nyquist of above 0.3. We perform three distinct tolerance runs: the first with only position errors, the second with only shape errors, the third with both position and shape errors at the same time.

4.3.2 Straylight analysis

We performed a systematic straylight analysis using OpticStudio in non sequential mode. Rudimentary baffling was placed where necessary, as shown in Fig. 4.9. All the straylight comes from a 90x76 mm window element with random angle within the hemisphere going into the sytem. The baffling is composed of two rectangular elements with 100 % absorption. No diffusive (nor diffraction *a fortiori*) straylight is considered and parts other than mirrors are ignored. We avoid obscuration entirely by placing baffling only outside of the nominal ray paths materialised on Fig. 4.10.

The Total Integrated Scatter (TIS) [161] (Eq. 4.1) is often used to estimate the fraction of incident flux that is reemitted diffusively at a diopter surface given the wavelength λ, the RMS surface roughness σ and the difference in refractive

Perturbation type	**Standard Deviation** (±)
Position only	
Piston	20 µm
X Decenter	25 µm
Y Decenter	25 µm
X Tilt	1′ (0.29 mrad)
Y Tilt	1′
Z Tilt (not the sensor)	1′
Shape only (Z4-Z45 Standard)	
RMS altitude error (all mirrors)	300 nm
Position & Shape	
Piston	15 µm
X Decenter	20 µm
Y Decenter	20 µm
X Tilt	0.78′ (0.23 mrad)
Y Tilt	0.78′
Z Tilt (not the sensor)	0.78′
RMS altitude error (all mirrors)	200 nm

Table 4.2: Tolerance analysis results. Perturbation standard deviations so that >90 % of drawn systems reach a mean diffraction MTF of > 0.3 at Nyquist.

index between incident and exit media Δn (2 for a mirror). In our case, we may estimate the TIS of our mirrors to be below 3.10^{-4} with the reasonable assumptions $\sigma = 10\,\text{nm}$, $\lambda = 8\,\mu\text{m}$. These levels of straylight are negligible at the design stage we are at.

$$\text{TIS} = \left(\frac{2.\pi.\Delta n.\sigma}{\lambda} \right)^2 \tag{4.1}$$

4.4 System Prescription Data

In this section, all the data necessary to replicate the YATMA optical design is given. We begin with the overall system geometry and then describe the shape of all the mirrors.

4.4.1 General prescription data

We present the prescription data for YATMA in Tab. 4.3. The shape of the mirrors is found further below (Tab. 4.4). The whole system is symmetric around the plane of the layout (Fig. 4.1). Except where otherwise noted, all length units are millimeters, all angles degrees. "CR" designates default OpticStudio *Coordinate Breaks*. Values are given with a sufficient numerical accuracy, a light optimization may be necessary to reproduce the exact nominal performance.

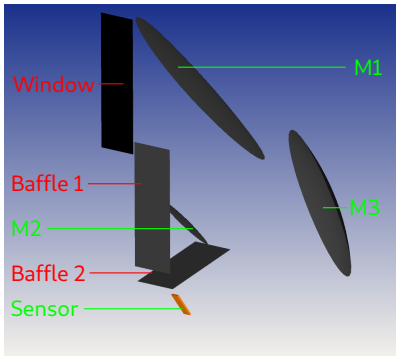


Figure 4.9: 3D view (OpticStudio) of the TMA and the rudimentary baffling that was used in the simulation.

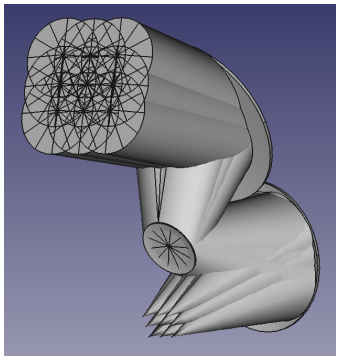


Figure 4.10: 3D CAD view of the TMA with the ray bundle solids for extremal fields. Mechanical parts placed outside the bundles do not generate obscuration.

#	Surface Type	Thickness	Decenter Y	Tilt X
OBJ	Standard	Inf	-	-
1	Standard	100	-	-
2	CR	-	-	-41.0713
3	M1			
4	CR	-77	-	-41.0713
5	CR	-	-	37.2007
6	M2 (STOP)			
7	CR	84.683	-	37.2007
8	CR	-	-	-12.6951
9	M3			
10	CR	-105.426	-	-12.6951
11	CR	-	-1.6054	-0.2298
IMA				

Table 4.3: YATMA prescription data: overall system geometry.

Parameter	M1	M2	M3
Semi-diameter (mm)	54	18	44.4
Base sphere curvature radius (mm)	-422.015	-119.037	-135.408
Zernike normalization radius (mm)	57	23	52
Z6 (mm)	-2,95853E-01	-8,7699E-02	1,2550E-02
Z7	4,3442E-02	7,50E-04	3,9146E-02
Z9	9,89E-04	-6,474E-03	-1,0893E-02
Z11	3,014E-03	-2,585E-03	-8,083E-03
Z12	4,885E-03	-4,83E-04	1,541E-03
Z14	7,98E-04	-1,52E-04	-1,096E-03
Z17	-2,99E-04	-1,24E-04	1,680E-03
Z19	-5,86E-04	-1,14E-04	-2,86E-04
Z22	3,3E-05	-2,0E-05	-3,77E-04
Z24	1,17E-04	-2,4E-05	1,9E-05
Z29	6E-06	(-1E-09)	5,6E-05
Z37	-1,3E-05	1,2E-05	-9E-06
Max. normal departure from base sphere (mrad)	36	16	13
Max. sag departure from base sphere (µm)	730	142	70

Table 4.4: Freeform mirrors shape description. Surface type: *Zernike Standard Sag*.

4.4.2 Mirrors

We give the complete description of the mirrors in Tab. 4.4. The coefficients are presented with the same signs as entered in OpticStudio and are consistent with the general system prescription data in Tab. 4.3. Zernike Standard coefficients are expressed in millimeters up to nanometer accuracy for conciseness. We use the OpticStudio indexing.

Additionally, let us plot some surface maps for the freeform mirrors to better assess the freeform complexity we are dealing with. For each mirror, we produce the following maps[1]:

- Sag. (Figs. 4.11, 4.15 and 4.18)
- Sag of the freeform component (base sphere subtracted). (Figs. 4.12, 4.13, 4.16 and 4.19)
- Angular departure of the freeform component (base sphere subtracted). Note this is the angle between the normal to the surface and the normal to the sphere, not the gradient (although they are related). (Figs. 4.14, 4.17 and 4.20). The figures are produced with a Zemax macro we have written.

[1]Please refer to the digital version of the present manuscript for high resolution pictures.

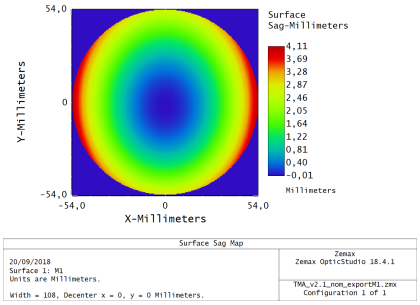


Figure 4.11: M1: Sag.

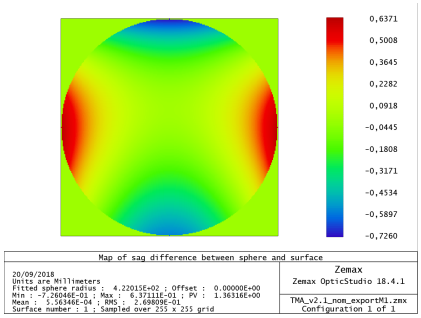


Figure 4.12: M1: Sag of the freeform component.

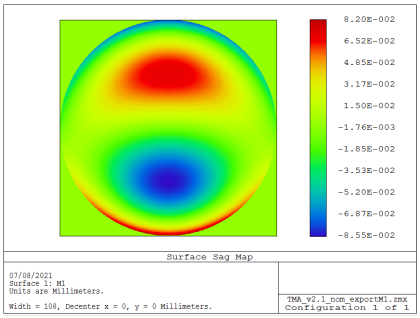


Figure 4.13: M1: Sag of the freeform component without astigmatism. Coma is the second dominant term.

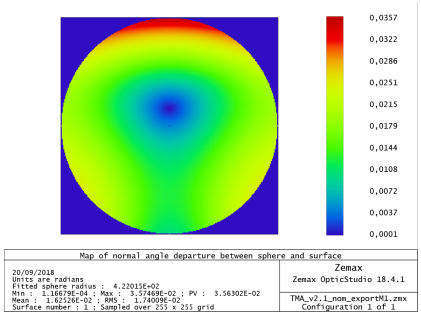


Figure 4.14: M1: Angular departure of the freeform component.

Note the freeform departure maps in sag and angle are produced using custom OpticStudio macros. Also note that for the maps below, the sign for the sag has been adapted so that positive altitude corresponds to bumps away from the face of the mirror (check the convexity/concavity of mirrors on the sag maps if there is still some ambiguity left).

Finally, we show the ray footprints on M1 and M2 for extremal fields (Figs. 4.21 and 4.22). This gives an idea of what the real useful area looks like and what part of the mirrors' faces could be used for eventual fiducials or could be removed altogether for weight reduction. Note that a small part at the bottom of M1 actually *needs* to be removed to avoid obscuration.

4.5 Discussion

The imaging performance of YATMA is satisfactory, with good near diffraction-limited MTF and good fastness. The design geometry is also quite compact. The distortion levels are manageable.

Although the exit pupil ellipticity was controlled in our merit function, it is

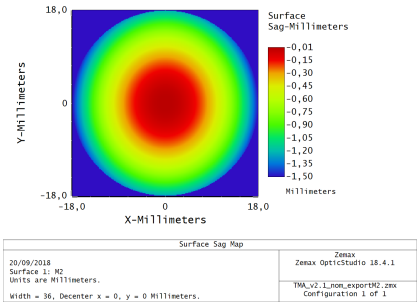


Figure 4.15: M2: Sag.

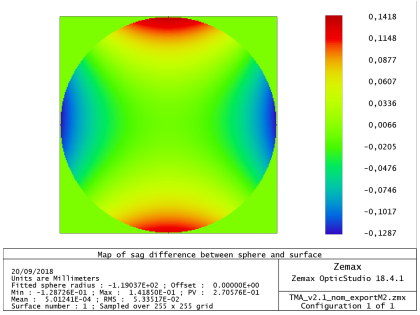


Figure 4.16: M2: Sag of the freeform component.

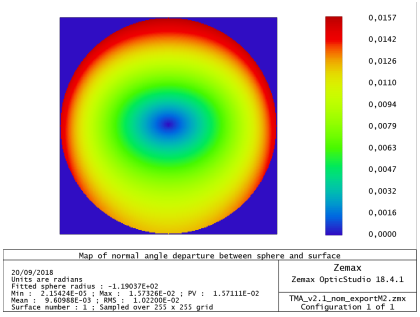


Figure 4.17: M2: Angular departure of the freeform component.

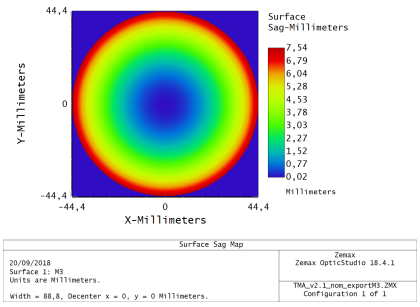


Figure 4.18: M3: Sag.

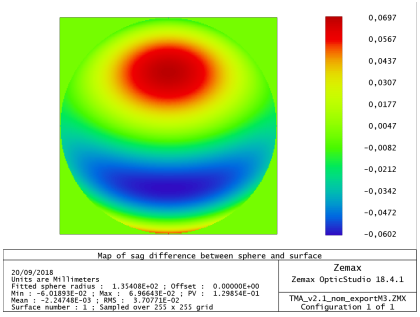


Figure 4.19: M3: Sag of the freeform component.

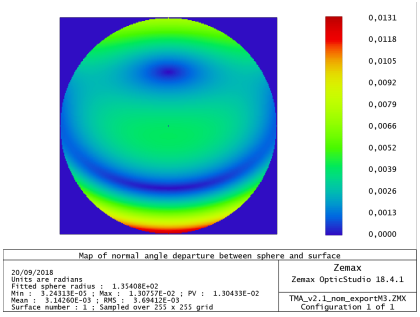


Figure 4.20: M3: Angular departure of the freeform component.

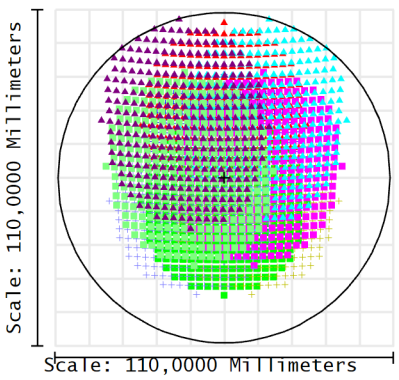


Figure 4.21: Footprint of extremal fields on M1. Note the clear aperture would be resized and shifted upwards for manufacturing.

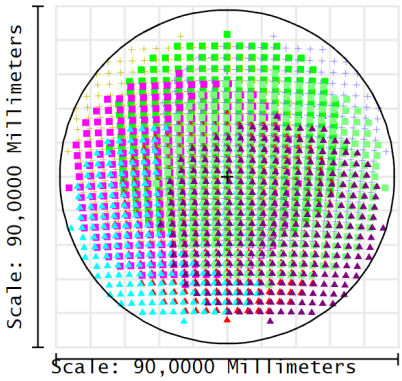


Figure 4.22: Footprint of extremal fields on M3.

however the limiting factor for image quality, we believe a better compromise can be found by degrading slightly the spot diagrams in the field in order to improve the exit pupil circularity. Also, the system needs baffling to avoid specular straylight, this is not common in TMA systems. Since we observe in thermal IR, this could prove problematic as the baffling is heated by the absorbed straylight and seen by the sensor.

The tolerancing shows the system is quite sensitive, although further studies could be carried out to better balance each error contributor. As far as manufacturing and surface shape metrology are concerned, some mirrors are quite challenging since they exhibit high slopes (M1 in particular with a maximum slope of 36 mrad[2]).

The primary focus of the CNES R&T FAME was assessing the manufacturing feasibility of a fast freeform mirror using ceramic processes. The fact that our M3 mirror is fast in our design could be detrimental to the optical performance, but our mission was to produce a voluntarily difficult case for the manufacturer as part of the research program. Likewise, the scope of the research program did not include manufacturing the whole system.

Our optical design led to the manufacturing of an unpolished M3 mirror using laser stereolithography (SLA) of ceramic material by Nicolas Rousselet at 3DCERAM (Figs. 4.23 and 4.24)[3]. This was done to assess the feasability of manufacturing a very fast mirror (M3 on its own has $N = 0.63$[4]). The advantages of this manufacturing method are:

- The freeform component is already included in the first manufacturing

[2]Though most of the freeform component is astigmatism so null systems could be used for metrology, as shown in [28]

[3]Actually the earlier version of M3 shown in [158], though the shape is close

[4]The mirror fastness was evaluated using its clear aperture and best focal length when used on its own with a collimated ray (see [158])

Figure 4.23: Front of the unpolished M3 mirror.

Figure 4.24: Back of the unpolished M3 mirror.

step. The designer is not constrained by the quantity of material that needs to be removed between the base sphere and the target freeform shape.

- The optical surface is in the same material and manufactured during the same step as the mechanical structure behind it. This eliminates *some* possibilities for positionning errors between the optical surface and the mechanical part.
- The part can be hollowed out as well as optimized for rigidity and vibration resistance.

Further characterization or manufacturing were not carried out for this design.

4.6 Acknowledgment

This research was financed as part of a CNES R&T program supervised by Vincent Costes.

We think the present chapter might be a good template of what an optical design preliminary study report for a freeform unobscured telescope should include. The design meets the requirements that were set at the time. However we spent a lot of time working on it with the means available to optical designers. Moreover, were the requirements to change enough to prompt a redesign, we would have to start over and likely spend a significant amount of time (minus the time spent creating the performance metrics, evaluations, literature review etc.) doing a rework. It is well known (especially to inexperienced optical designers) that finding the right optical design, or at least the one that even begins to provide good performance, is akin to finding a needle in a haystack. Motivated by our newfound experience in freeform optical design, we have investigated ways

to search haystacks faster using search algorithms. In order to investigate the matter with total control over the computational flow, we have first implemented tools for sequential raytracing, as will be explained in the next two chapters.

Chapter 5

Sequential raytracing

Sequential raytracing vs other types of raytracing – Coordinate systems – Ray propagation – Sphere intersections – Surface normals – Snell-Descartes reflection and refraction – Error cases – Gratings – General (freeform) surface intersection – Our implementation: validation and performance – Parallelism – GPU – Software architecture – Advice to implementers

Motivated by understanding the exact computational flow behind what is called *sequential raytracing* in commercial software, we have implemented our own tools to perform the task. Publishing (see below) on sequential raytracing seem to have been more popular some decades ago than it is now. This is perhaps due to the fact that commercial software have largely solved the issue for most purposes. The unfortunate consequence being that there is not a huge deal of resources on the details of doing such computations. The following chapter covers topics in sequential raytracing. It is organized as a template for actually programming a sequential raytracer, as closed-form formulae are provided whenever possible and reasonings on how to solve common issues are given.

Firstly we remind basic concepts such as Snell-Descartes refraction and reflection, then we move onto more advanced topics such as simulating gratings and computing the intersection of rays with freeform surfaces, lastly we give some test results of our own implementation and advice on the questions that arose concerning the software architecture during development.

5.1 Context

Sequential raytracing for optical design is the application of geometric ray propagation through a known sequence of diopters and media from an object space to an image space. It satisfies constraints of predefined sampling of field or aperture. It constitutes the core simulation method of optical performance for the largest part of imaging systems.

Several related fields of study exist, which must not be confused with sequential raytracing. First, so-called *non-sequential* raytracing is the propagation of

rays through an *a priori unknown* sequence of diopters and media. Rays emerge from light sources towards objects that interact with light and finally are eventually absorbed[1]. Applications are mainly lighting and straylight analysis. This type of simulation typically mixes optical components with the surrounding mechanical components and sources and lights. It takes into account, in most cases, a scattering model for diopters, surfaces and volumes. It outputs data on which object were struck by light, how much light was absorbed *etc.* Despite involving the same surface types as sequential raytracing, non-sequential raytracing is fundamentally different: the ray propagation is made substantially harder by the need to find the next object to interact with for each ray.

Another related field, much more popular, is raytracing for visual rendering (*eg* for computer games and animated movies). This field has much more in common with non-sequential raytracing than sequential raytracing. The need for realism in visual rendering drives the inclusion of more and more rigorously implemented physical phenomena in raytracing engines, to the point that using engines originally made for visual rendering in actual physics simulation has become viable [163, 164].

Also note that the raytracing we are concerned with here is referred to as *real* raytracing, *ie* the application of Snell-Descartes laws at diopters for every ray instead of the propagation of quantities like the Lagrange invariant via the raytracing of one ray. We are also not concerned with physical optics (propagation of electromagnetic fields using *eg* Fresnel propagation and Fourier optics)[2].

Several references exist on real raytracing [166–170]. In the present chapter, we describe some low-level mathematical descriptions for raytracing but omit code optimization and numerical accuracy considerations (which are nonetheless important for fast and accurate raytracing implementation).

5.2 Basic Concepts

Let us present sequential raytracing concepts. We will only brush the concepts and do not detail their implementation. The building blocks we give are all the computations needed to build a simple raytracer (omitting ray-aiming for now, which is the hard part). As we'll see, the mathematics of the problem are rather straightforward.

5.2.1 Ray representation

The fundamental data object for real raytracing is the ray representation. A simple, viable representation is composed of a point xyz and a unit vector lmn in the direction of light propagation (Tab. 5.1).

[1] See for instance [162] for the implementation of non-sequential computations.
[2] See for instance [165] for a treatment of the subject of implementing such programs.

x	
y	
z	*eg* the altitude of the intersection on a surface at (x, y)
l	x direction cosine
m	y direction cosine
n	z direction cosine

Table 5.1: Ray data representation.

Optionally, optical path value, polarization, power *etc* can be added to this representation for specialized simulations[3].

5.2.2 Coordinate systems and transfer

We need a way to propagate rays from one optical surface to the next in homogeneous media. We choose to represent optical systems (simple ones in any case) as a succession of optical surfaces which typically apply intersections and refraction/reflection on a ray, separated by propagation in straight lines from one surface to the next.

Contrary to non-sequential raytracing, there is little to gain in sequential raytracing by adopting a global coordinate system, it can, to the contrary, cause numerical accuracy issues. We rather choose, along with commercial software (as far as we know), to adopt a local coordinate system relative to each optical surface. Optical systems are a succession of optical surfaces separated by coordinate *transfer* from one surface to the next. A necessary operation is then to transform the ray coordinates $xyzlmn$ from one coordinate system to the next, given *eg* a rotation matrix P_{trans} and a translation vector (DX, DY, DZ) (or any representation for *rigid motion*) between the two (Fig. 5.1). This can be represented for the user as a sequence of tilts and translations along each coordinate axis[4]. This *transfer* from one surface coordinate system to the next is the equivalent of *Coordinate Break* in OpticStudio. This raytracing concept is key to the design of off-axis systems (such as unobscured telescopes).

We illustrate this *transfer* concept from one surface coordinate system to the next with Fig. 5.2[5]. We *transfer* rays from their position P_1 in the initial coordinate system (for example just after applying an intersection with the surface and a refraction), with initial direction $\overrightarrow{d}$ to the next coordinate system, rotated and with its origin at the next apex A_2. The point P_2 is the intersection

[3]Note additionally that the *de facto* standard for number representation in raytracing is currently IEEE double-precision floating point number. The difference between single and double precision is important for accuracy in many optical systems (nanometer intersection precision over meters of propagation repeated for whatever number of optical surfaces there are). In visual rendering single-float numbers are traditionally used, but with the advent of General Purpose GPU (GPGPU) double-float computing was introduced on most modern graphics cards.

[4]The actual order of transformations matters little as long as it is indicated clearly.

[5]Note the coordinate system is with the Z axis opposed to the direction of light but it is just a matter of convention, the computation remains fundamentally the same.

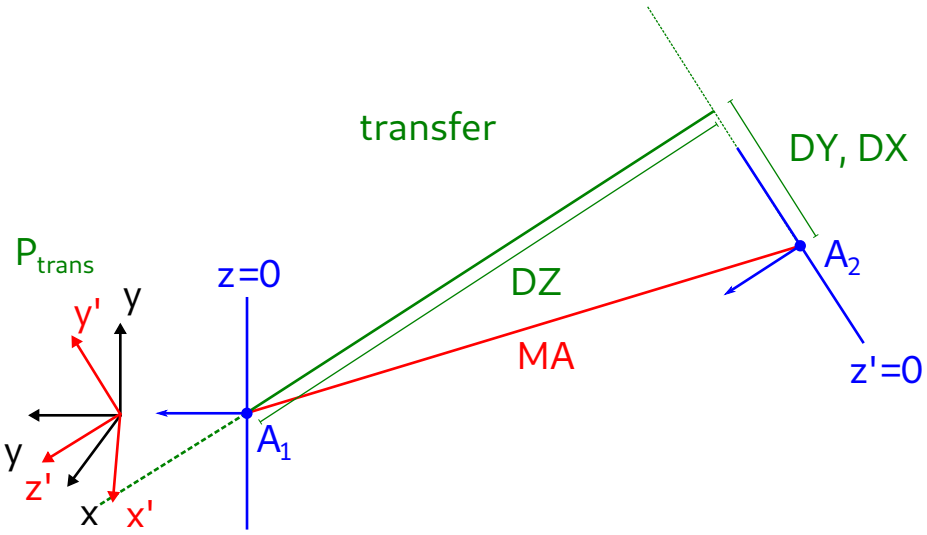


Figure 5.1: Quantities characterizing a *transfer* between two successive optical surfaces. A_1 and A_2 are the apex of the two successive optical surfaces, joined by the mechanical axis (MA). A $z = 0$ plane goes through each apex. The unprimed coordinate system has origin A_1 and pertains to the first surface. The primed coordinate system has origin A_2 and pertains to the second optical surface.

of the ray with the $z' = 0$ plane. This choice aligns all the rays on a plane, and makes the subsequent intersection with the next surface easier to perform.

For the sake of completeness, let us give the mathematical expressions needed to compute the ray intersection from one surface coordinate system to the next.

The rotation matrix from the initial coordinate system to the next is such that it links lmn and $l'm'n'$ as in Eq. 5.1[6].

$$\begin{bmatrix} l' \\ m' \\ n' \end{bmatrix} = P_{\text{trans}}^{-1} \cdot \begin{bmatrix} l \\ m \\ n \end{bmatrix} \tag{5.1}$$

P_1 can be expressed in the new coordinate system with Eq. 5.2.

$$\begin{bmatrix} x_1' \\ y_1' \\ z_1' \end{bmatrix} = P_{\text{trans}}^{-1} \cdot \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} - \begin{bmatrix} DX \\ DY \\ DZ \end{bmatrix} \tag{5.2}$$

And only the intersection with the $z' = 0$ plane is left to obtain the coordinates of P_2 in the primed coordinate system (Eq. 5.3)[7].

[6] P_{trans}^{-1} can of course be pre-computed (and is, in our implementation). The same goes for any quantity independent from ray coordinates.

[7] Note it is actually possible to miss the plane if the ray is parallel to it. This is a special raytracing case.

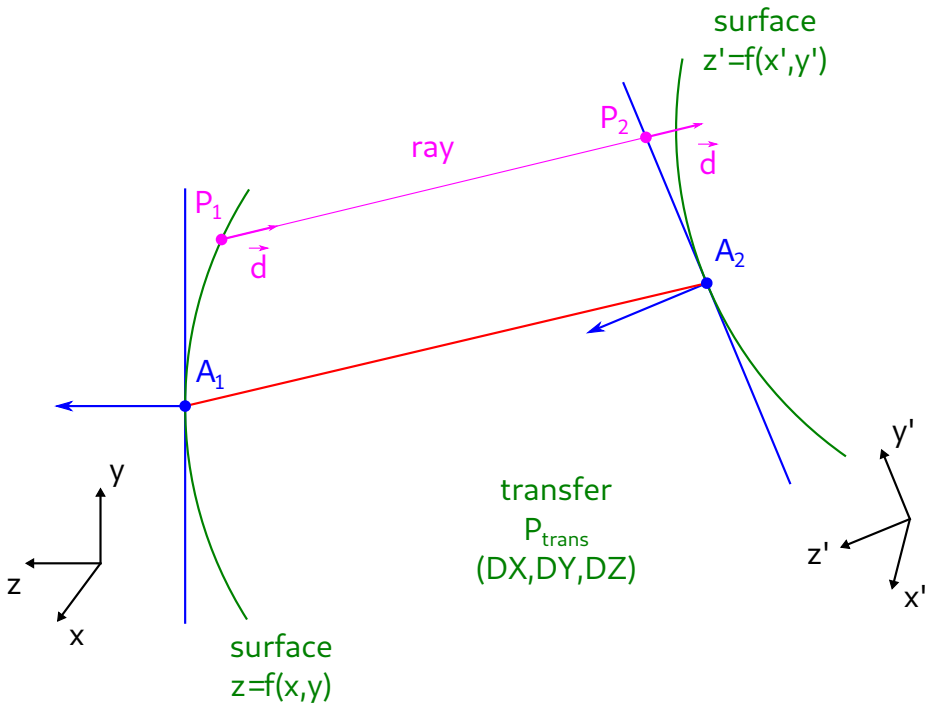


Figure 5.2: Diagram for coordinate system transfer from one surface to the next. The ray coordinates $xyzlmn$ starting on the previous surface at point P_1 in the coordinate system A_1xyz are simply expressed in the new coordinate system $A_2x'y'z'$ and the intersection P_2 is found at the new $z' = 0$ plane.

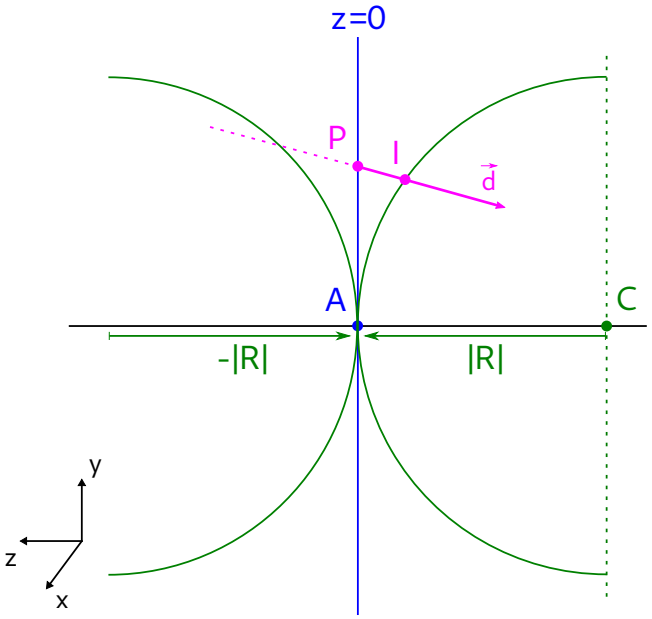


Figure 5.3: Ray intersection with spheres diagram. Here I is on a sphere with positive R, the computations are exactly the same with spheres with negative R.

$$\begin{bmatrix} x'_2 \\ y'_2 \\ z'_2 \end{bmatrix} = \begin{bmatrix} x'_1 \\ y'_1 \\ z'_1 \end{bmatrix} + t \cdot \begin{bmatrix} l' \\ m' \\ n' \end{bmatrix} = \begin{bmatrix} x'_1 - \frac{z'_1}{n'} l' \\ y'_1 - \frac{z_1}{n'} m' \\ 0 \end{bmatrix} \tag{5.3}$$

Functionally, the *transfer* operation, for a given ray, computes the numbers $(x'_2, y'_2, 0, l', m', n')$ from the previous ray (x_1, y_1, z_1, l, m, n) and P_{trans}, (DX, DY, DZ)[8].

5.2.3 Surface intersection: basic sphere case

We need to compute the intersection between rays and spheres. Thankfully, this is a problem for which the solution is a closed-form (we'll see in Section 5.3.2 the general case for more complex surface shapes).

Starting with a ray at P with direction $\overrightarrow{d}$, we want to compute I the point of intersection with the sphere of apex A, center C and radius R (signed) (Fig. 5.3).

The points along the ray obey Eq. 5.4.

[8] Readers might wonder why we express rotations with rotation matrices rather than with quaternions, as is usual in computer graphics. The reason is that rotating vectors is twice as fast with rotation matrices as with quaternions. We also do not need to chain rotations at execution time in our implementation and we do not encounter *gimbal lock* issues.

$$\begin{bmatrix} x(t) \\ y(t) \\ z(t) \end{bmatrix} = \begin{bmatrix} x_P + tl \\ y_P + tm \\ tn \end{bmatrix} \tag{5.4}$$

The points of the sphere obey Eq. 5.5.

$$x^2 + y^2 + (z - R)^2 = R^2 \tag{5.5}$$

Combining the two and knowing that $l^2+m^2+n^2 = 1$, we obtain a quadratic equation $t^2 + bt + c = 0$ with:

$$\begin{cases} b = 2(x_P \cdot l + y_P \cdot m - nR) \\ c = x_P^2 + y_P^2 \\ \Delta = b^2 - 4c \end{cases} \tag{5.6}$$

We want the intersection closest to the $z = 0$ plane out of the eventual two, so we choose the solution in Eq. 5.7.

$$t_{\text{sol}} = \frac{-b + \text{sign}(b)\sqrt{\Delta}}{2} \tag{5.7}$$

Finally, the intersection (if it exists) is given by Eq. 5.8.

$$\begin{bmatrix} x_I \\ y_I \\ z_I \end{bmatrix} = \begin{bmatrix} x_P + t_{\text{sol}} \cdot l \\ y_P + t_{\text{sol}} \cdot m \\ t_{\text{sol}} \cdot n \end{bmatrix} \tag{5.8}$$

The problem is not so simple as it might first appear, indeed we have special cases to manage during the raytracing, these are, in order:

1. $\Delta < 0$: No intersection between the ray and the sphere exists.
2. $|z_I| > |R|$: This is an intersection beyond the hemisphere closest to the $z = 0$ plane. The math works out, but this is not what we want to define with spherical surface types.

We will encounter other special cases below, we summarize them in Section 5.2.7.

5.2.4 Surface Normal

Computing the surface normal (or first derivatives) is required for Snell-Descartes laws. Let's review the spherical case and then the general one.

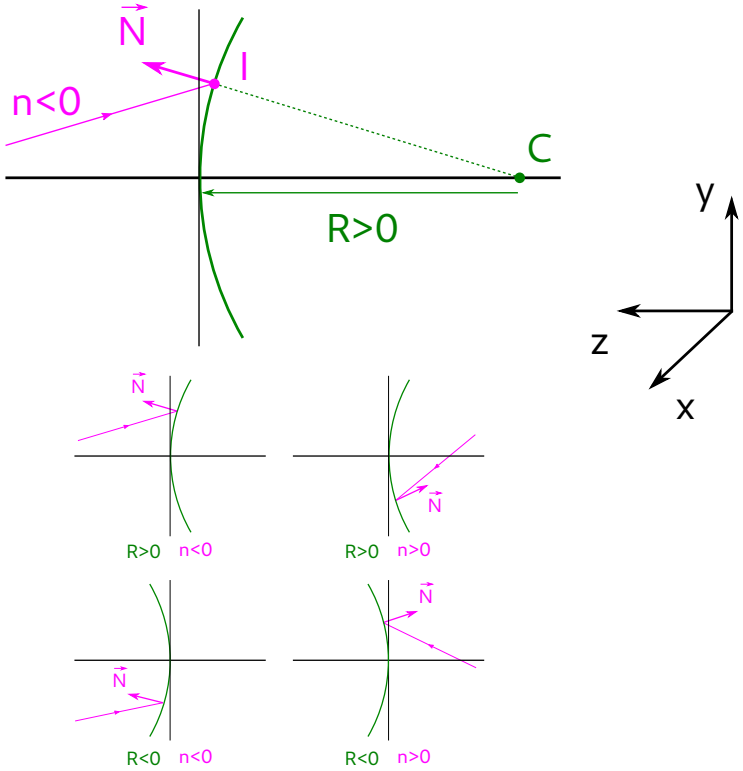


Figure 5.4: Sphere normal vector N at I.

5.2.4.1 Sphere normal vector

Knowing the point of intersection I on the sphere, we want to compute the corresponding unit normal vector $\overrightarrow{N}$. The trick is we always want the vector pointing in the direction opposed to that of the incoming rays[9]. We illustrate this in Fig. 5.4.

$\overrightarrow{CI}$ (Eq. 5.9) is colinear to $\overrightarrow{N}$.

$$\overrightarrow{CI} = \begin{bmatrix} x_I \\ y_I \\ \text{sign}(R)\sqrt{R^2 - (x_I^2 + y_I^2)} \end{bmatrix} \tag{5.9}$$

$\overrightarrow{CI}$ normalized and flipped opposite to the incoming ray gives $\overrightarrow{N}$ (Eq. 5.10).

$$\overrightarrow{N} = -\text{sign}(n) \cdot \text{sign}(R) \cdot \frac{\overrightarrow{CI}}{|R|} = -\text{sign}(n) \cdot \begin{bmatrix} x_I/R \\ y_I/R \\ \sqrt{R^2 - (x_I^2 + y_I^2)}/|R| \end{bmatrix} \tag{5.10}$$

[9] This is a choice on our part to indicate which side of the surface is being intersected by the ray. This is useful for mirror reflection (Section 5.2.5).

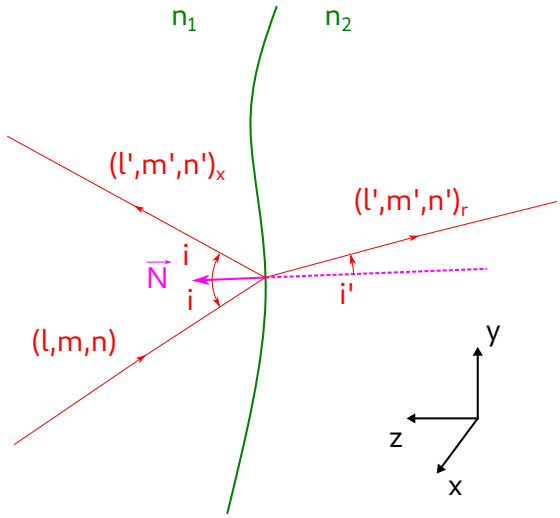


Figure 5.5: Snell-Descartes law illustration: Incident, refracted (r) and reflected (x) rays in the plane of incidence on an arbitrary surface.

5.2.4.2 Normal vector: general case

The unit normal vector $\overrightarrow{N}$ can be computed easily from the first spatial derivatives in the X and Y directions (Eq. 5.11), with $z = f(x, y)$ the function computing the altitude of the surface [171].

$$\overrightarrow{N} = \frac{\text{sign}(n)}{\sqrt{1 + \left(\frac{\partial f}{\partial x}(x,y)\right)^2 + \left(\frac{\partial f}{\partial y}(x,y)\right)^2}} \cdot \begin{bmatrix} \frac{\partial f}{\partial x}(x,y) \\ \frac{\partial f}{\partial y}(x,y) \\ -1 \end{bmatrix} \tag{5.11}$$

5.2.5 Snell-Descartes refraction and mirrors

Given an incident ray with direction lmn and a unit surface normal $\overrightarrow{N}$. We can compute the new direction $l'm'n'$ of the ray in the refraction case (between media with refraction indices n_1 and n_2) and in the reflection case (Fig. 5.5) [166, 167].

First we need to compute the angle of incidence (its cosine) via the scalar product between ray direction and normal (Eq. 5.12).[10]

$$\cos(i) = \overrightarrow{lmn} \cdot \overrightarrow{N} \tag{5.12}$$

[10] Note $\cos(i)$ is always negative since the incident direction and $\overrightarrow{N}$ are always in opposite half planes.

5.2.5.1 Reflection case

The new direction in reflection is simply given by subtracting twice the scaled normal from the incident vector Eq. 5.13.[11]

$$\overrightarrow{(l'm'n')_x} = \overrightarrow{lmn} - 2\cos(i) \cdot \overrightarrow{N} \tag{5.13}$$

5.2.5.2 Refraction case

The scalar law of refraction $n_1 \cdot \sin(i) = n_2 \cdot \sin(i')$ gives Eq. 5.14.

$$\cos(i') = \sqrt{1 + \left(\frac{n1}{n2}\right)^2 \left(\cos(i)^2 - 1\right)} \tag{5.14}$$

We then have the new refracted ray direction $\overrightarrow{(l'm'n')_r}$ (Eq. 5.15).

$$\overrightarrow{(l'm'n')_r} = \frac{n1}{n2} \cdot \overrightarrow{lmn} - k \cdot \overrightarrow{N} \tag{5.15}$$

With:

$$k = \frac{n1}{n2} \cdot \cos(i) + \cos(i') \tag{5.16}$$

5.2.5.3 Special case: TIR

We have a case of Total Internal Reflection (TIR) when $1 + \left(\frac{n1}{n2}\right)^2 (\cos(i)^2 - 1) < 0$.

5.2.6 Sequence application on a ray

For a given ray and a given sequence of diopters, the propagation of the ray from the start of the sequence to its end can be seen as the successive application of:

1. transfer (Section 5.2.2)
2. intersection (Section 5.2.3)
3. normal computation (Section 5.2.4)
4. snell refraction/reflection (Section 5.2.5)
5. transfer
6. etc.

Fig. 5.6 summarizes these steps and the involved data.

[11] Note that, with our definition of $\overrightarrow{N}$, this automatically points the ray in the direction of the propagation of light. We flip the ray and not the coordinate system (as is sometimes legitimately done with mirrors).

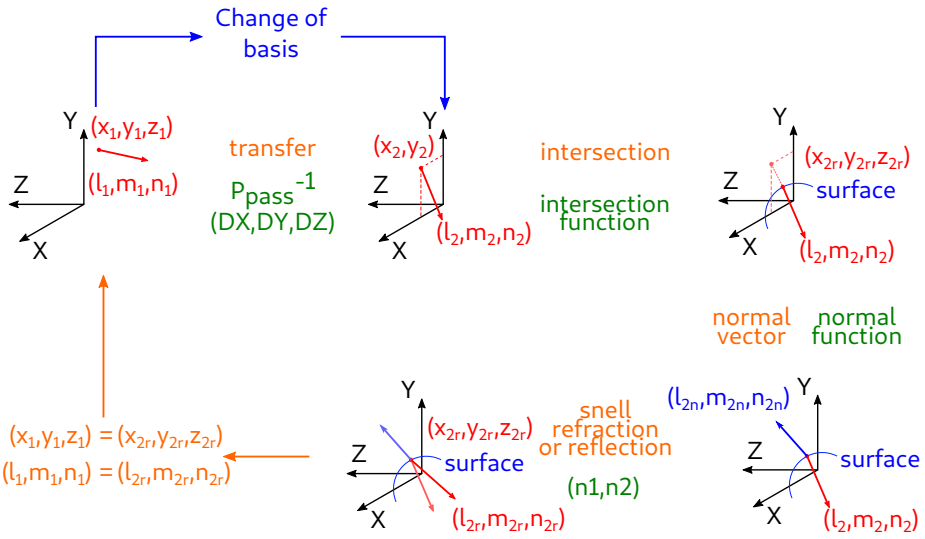


Figure 5.6: General raytracing ray propagation through a sequence of diopters.

5.2.7 Special cases and errors

Fig. 5.7 summarizes the different special cases and errors we encounter in basic raytracing. These are:

- TIR (Section 5.2.5).
- Ray missing the surface (Section 5.2.3).
- Intersecting a sphere beyond the intended hemisphere (Section 5.2.3).

Some control flow is necessary to manage these special cases during raytracing. We can either signal an error which interrupts the current MF computation, or drop the rays with errors and carry on with the rest of the raytracing with the remaining rays.

5.3 Advanced topics

The above description is sufficient to implement a raytracer that propagates rays from an entrance pupil to an image plane through a sequence of spherical diopters, possibly off-axis and with 3D geometry.

Let us review two more advanced concepts in raytracing: gratings and general ray/surface intersections.

5.3.1 Gratings

As an example of a *special* surface, let us look at raytracing through gratings [166, 167, 172, 173]. We give the completely general treatment of a freeform

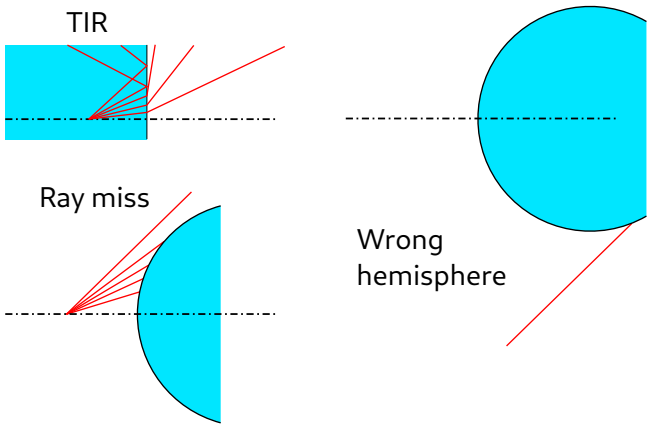


Figure 5.7: Summary of special cases in basic raytracing.

phase function on a freeform substrate[12]. Such gratings are useful for instance in spectro-imager systems, see [174] for a freeform substrate grating used in spectro-imagers designs. As we'll see, they integrate the already presented computation flow, the intersection function and the function computing the output ray direction are simply replaced.

In the case of gratings raytracing, we have to ignore many diffraction effects. We only raytrace a single order of diffraction and ignore polarisation, scattering and efficiency. Our idealized grating is described by a substrate shape and a phase function. The raytracing model is:

1. Intersect the ray and substrate (see Section 5.3.2). This gives a point I of intersection. We also compute the unit normal vector $\overrightarrow{N}$ to the substrate as usual.

2. Determine the groove direction and step at the intersection point from the phase function $\phi(x, y)$. Locally, the effect of the grating on the ray is the same as that of a straight grating with these direction and step size.

3. Compute the output ray direction using the vectorial grating equation.

We consider the phase function $\phi(x, y)$ a given, defined by the optical designer[13]. The usual treatment is to define the grooves as the intersection between the substrate and iso-surfaces in a 3D phase field (typically interference fringes) [172]. For practical purposes, we can use without loss of generality a 3D phase field constant in the z direction $\phi(x, y)$ since this is how we propose to define the phase (rather than with two interfering sources) (see Fig. 5.9). This simplifies the phase definition as well as raytracing computations.

[12] Although this might be difficult to manufacture
[13] The 2D phase can be defined by freeform polynomials for instance.

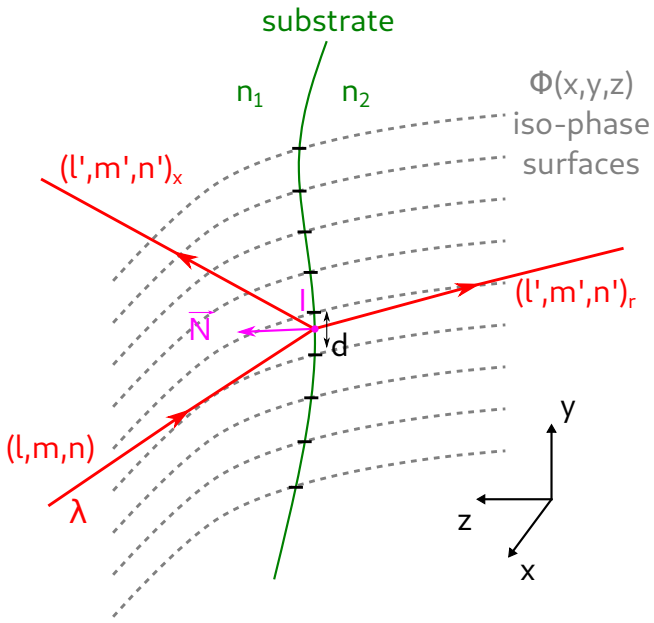


Figure 5.8: Grating raytracing notations. The intersection of the iso-phase curves with the substrate define the grooves positions. The grooves on the substrates are the little black lines.

For future reference, let us give the whole derivation for the raytracing equations, in the general case (with a 3D phase $\phi(x, y, z)$). This derivation is contained in [167, 172]. Figs. 5.8 and 5.10 illustrate some of the notations we use. When possible we take the same notations as Rayces [172] but we maintain consistency with the preceding paragraphs. Let us make a list:

- n_1, n_2: Respectively incident and exit refraction indices.
- $\overrightarrow{lmn}$, $\overrightarrow{l'm'n'}_r$, $\overrightarrow{l'm'n'}_x$: Incident, transmitted and reflected ray directions on or through the grating.
- λ: Vacuum wavelength of the incident ray.
- $\phi(x, y, z)$: General 3D phase function for the grating.
- d: *Local* groove step (in units of length).
- $\overrightarrow{N}$: Unit normal vector to the substrate at intersection I with the ray, defined as before to be in half-plane opposite to $\overrightarrow{lmn}$.
- $\overrightarrow{\nabla\phi}$: phase gradient.
- $\overrightarrow{s}$: unit vector in the plane of $\overrightarrow{N}$ and $\overrightarrow{\nabla\phi}$, tangent to the substrate and orthogonal to the grooves direction.

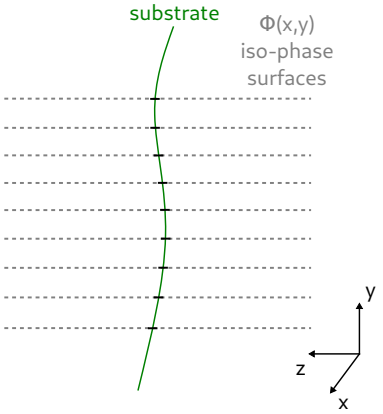


Figure 5.9: From a given set of grooves, we can always define a phase function constant in the z direction that generates the grooves. We simply extend the grooves in straight surfaces in the z direction.

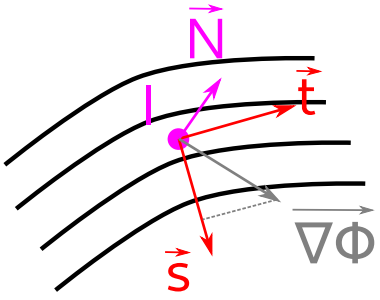


Figure 5.10: Illustration for the vectors used in the grating equation. Inspired from Fig.2 in [172].

- $\overrightarrow{t}$: unit vector parallel to the grooves direction, we have $\overrightarrow{N} \times \overrightarrow{s} = \overrightarrow{t}$.
- m_d: Diffraction order.

The *implicit* raytracing equation for gratings, both in transmission and reflection is Eq. 5.17.

$$\overrightarrow{l'm'n'} \times \overrightarrow{N} = \mu \cdot \overrightarrow{lmn} \times \overrightarrow{N} + \Lambda \cdot \overrightarrow{t} \tag{5.17}$$

With:

$$\begin{cases} \mu = \frac{n_1}{n_2} \\ \Lambda = \frac{m_d \cdot \lambda}{n_2 \cdot d} \end{cases} \tag{5.18}$$

Observing that $\overrightarrow{t} = -\overrightarrow{s} \times \overrightarrow{N}$, we can find a factorization (Eq. 5.19).

$$\left(\overrightarrow{l'm'n'} - \mu \cdot \overrightarrow{lmn} + \Lambda \cdot \overrightarrow{s}\right) \times \overrightarrow{N} = \overrightarrow{0} \tag{5.19}$$

This means the quantity multiplied by $\overrightarrow{N}$ is zero plus some vector $\Gamma \cdot \overrightarrow{N}$ (Eq. 5.20). This is the *explicit* grating raytracing equation. The same in scalar format is Eq. 5.21, with $\overrightarrow{s} = (u, v, w)$.

$$\overrightarrow{l'm'n'} = \mu \cdot \overrightarrow{lmn} - \Lambda \cdot \overrightarrow{s} + \Gamma \cdot \overrightarrow{N} \tag{5.20}$$

$$\begin{cases} l' = \mu \cdot l - \Lambda \cdot u + \Gamma \cdot l_N \\ m' = \mu \cdot m - \Lambda \cdot v + \Gamma \cdot m_N \\ n' = \mu \cdot n - \Lambda \cdot w + \Gamma \cdot n_N \end{cases} \tag{5.21}$$

We can see, by the way, that the sign of $\overrightarrow{s}$ does not really matter, as it will just swap negative refraction orders with positive ones.

We now need to compute the missing quantities to plug into Eq. 5.20:

- $\overrightarrow{s}$ (Eq. 5.23) and d (Eq. 5.24).
- Γ (Eq. 5.28)

Observe that:

$$\overrightarrow{t} = \pm \frac{\overrightarrow{\nabla\phi} \times \overrightarrow{N}}{\left\| \overrightarrow{\nabla\phi} \times \overrightarrow{N} \right\|} \tag{5.22}$$

We now have $\overrightarrow{s}$ (Eq. 5.23).

$$\overrightarrow{s} = \overrightarrow{t} \times \overrightarrow{N} = \pm \frac{\overrightarrow{\nabla\phi} \times \overrightarrow{N}}{\left\| \overrightarrow{\nabla\phi} \times \overrightarrow{N} \right\|} \times \overrightarrow{N} \tag{5.23}$$

The groove density $\frac{1}{d}$ is simply the length of phase gradient projected on $\overrightarrow{s}$ (Eq. 5.24).

$$\overrightarrow{\nabla\phi} \cdot \overrightarrow{s} = \frac{1}{d} \tag{5.24}$$

We now have all the numerical data to compute Λ. Let us move onto the matter of computing Γ. Squaring and adding all the lines in Eq. 5.21, we obtain Eq. 5.25.

$$\begin{aligned} 1 =& \mu^2 + \Lambda^2 + \Gamma^2 \\ & - 2\mu \cdot \Lambda (l \cdot u + m \cdot v + n \cdot w) \\ & + 2\mu \cdot \Gamma (l \cdot l_N + m \cdot m_N + n \cdot n_N) \\ & - 2\Lambda\Gamma (u \cdot l_N + v \cdot m_N + w \cdot n_N) \end{aligned} \tag{5.25}$$

If we observe that $\overrightarrow{s} \cdot \overrightarrow{N} = 0$, we obtain Eq. 5.26, a quadratic equation in Γ.

$$\Gamma^2 + \Gamma \cdot b + c = 0 \tag{5.26}$$

With:

$$\begin{cases} b = 2\mu \cdot \overrightarrow{lmn} \cdot \overrightarrow{N} \\ c = \mu^2 + \Lambda^2 - 1 - 2\mu\Lambda \cdot \overrightarrow{lmn} \cdot \overrightarrow{s} \\ \Delta = b^2 - 4c \end{cases} \tag{5.27}$$

We solve it, there are solutions iff $b^2 \geq 4.c$. For instance, there are no solutions once the step d starts becoming too small compared to λ.

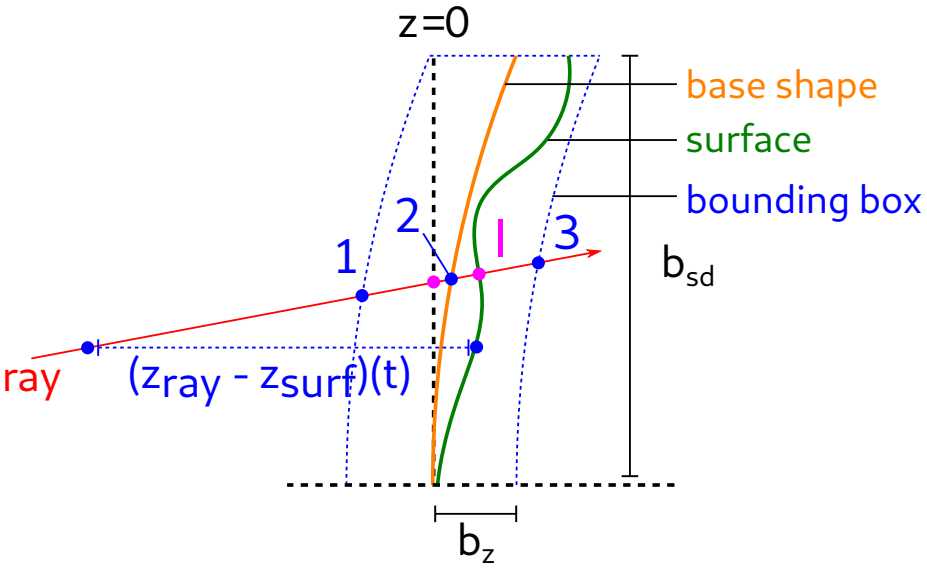


Figure 5.11: General ray/surface intersection problem illustration. The intersection to be found is point I. Points 1, 2 and 3 are easily accessible points where the altitude of the surface can be evaluated, they are 1: Start of the bounding box, 2: Base shape intersection, 3: Exit of the bounding box.

Our two roots give us the solutions in transmission and in reflection (Eq. 5.28).

$$\begin{cases} \Gamma = \frac{-b-\sqrt{\Delta}}{2} & \text{transmission} \\ \Gamma = \frac{-b+\sqrt{\Delta}}{2} & \text{reflection} \end{cases} \tag{5.28}$$

To determine which root is which, in our raytracing formalism with $\overrightarrow{N}$ always in the opposite half-plane to $\overrightarrow{lmn}$, observe in Eq. 5.20 that positive Γ pulls the exit ray direction towards $\overrightarrow{N}$. We also see that $-b = -2\mu \cdot \overrightarrow{lmn} \cdot \overrightarrow{N} > 0$. The *reflection* root is then the one that pulls the exit $\overrightarrow{l'm'n'}$ the most towards $\overrightarrow{N}$, or the largest of the two roots.

5.3.2 Surface intersection: general case

As we have seen (Section 5.2.3), the problem of intersecting rays with spheres has a solution in closed-form. This is not the case in general for more complex optical surfaces such as aspheres and freeform, even though they might have a polynomial expression. The problem of intersecting rays with general surfaces has to be solved iteratively [166] and is rather open-ended. We provide useful general tools to tackle the problem but leave the detailed case by case implementation up to the reader.

The problem is illustrated by Fig. 5.11. We want to determine I the intersection between the ray and the surface. We only have equations for the altitude and, in most cases, first derivatives of the surface at (x, y): $z_{\text{surf}}(x, y)$ and $\left(\frac{\partial z_{\text{surf}}}{\partial x}(x, y), \frac{\partial z_{\text{surf}}}{\partial y}(x, y)\right)$.

To solve the intersection problem, the general idea is to express it as a 1D root-finding problem, which are a particular subset of search problems[14], solvable by well-known algorithms such as the bisection method, Newton-Raphson or TOMS 748 [92]. We can express the intersection problem as a 1D root-finding problem readily (Eq. 5.29). The choice of root-finding algorithm depends on the availability of the derivative (they exist for all the surface types we have shown in the present work).

$$f(t) = z_{\text{ray}}(t) - z_{\text{surf}}(x(t), y(t)) \tag{5.29}$$

With the ray parametrized with t as usual, from the intersection with the $z = 0$ plane $(x_0, y_0, 0)$ as explained in Section 5.2.2 (Eq. 5.30).

$$\begin{cases} x_{\text{ray}}(t) = x_0 + t \cdot l \\ y_{\text{ray}}(t) = y_0 + t \cdot m \\ z_{\text{ray}}(t) = 0 + t \cdot n \end{cases} \tag{5.30}$$

The derivative $f'(t)$ will be given by Eq. 5.31.

$$f'(t) = n - \left(l \cdot \frac{\partial z_{\text{surf}}}{\partial x}(x, y) + m \cdot \frac{\partial z_{\text{surf}}}{\partial y}(x, y)\right) \tag{5.31}$$

We use the concept of bounding box, which is usual in visual rendering and can be found in optical raytracing for instance in Morita [104]. The general idea is to achieve *bracketing* on $f(t)$, that is find t_a and t_b such that $f(t_a) < 0 < f(t_b)$. By virtue of the continuity of the surface shape, we are then assured of finding t_i (at least one) such that $f(t_i) = 0$ somewhere between t_a and t_b. By defining a sufficiently large bounding box (compared with the freeform altitude departure) around *eg* the base shape of a freeform surface we can then compute $f(t)$ at the entrance and exit of the bounding box (points 1 and 3 on Fig. 5.11) and hopefully reach a bracket. The bounding box we propose on Fig. 5.11 is a shift of the base shape by b_z in either z directions. It is closed off radially by a maximum semi-diameter of b_{sd}[15]. Point 2 (on the intersection with the base shape) can also be useful to find a smaller bracket (starting the 1D search with a smaller bracket obviously makes for a faster search). In cases where no bracketing can be found, we have to give up and mark the ray as failed/invalid or have recourse to other methods such as ray marching (see below) at the cost of speed.

[14] Search algorithms crop up in many places in sequential raytracing, not just in the MF optimization.

[15] Many other bounding boxes could be proposed, for instance a simple cuboid around the apex, though managing its size could prove troublesome. A good bounding box balances the probability of achieving bracketing (robustness) with the size of the bracket (speed).

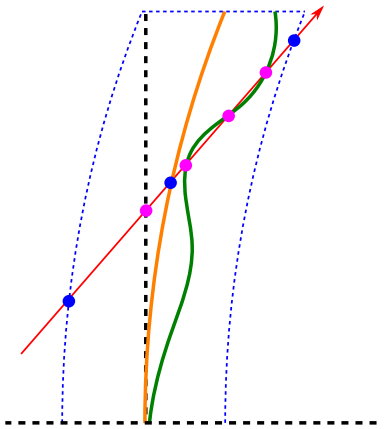


Figure 5.12: The bracketting is successful but there are several roots of $f(t)$ within it. The correct one is the one with smallest t in our formalism.

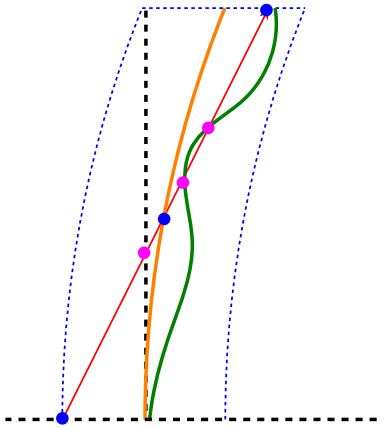


Figure 5.13: The bracketting fails! All the points used for the bracket (in blue) are on the same side of the surface ($f(t)$ is always the same sign).

We do not give a "one size fits all" solution to the problem because many factors come into play. The strategy we have outlined works *most of the time* for reasonable freeform departures. Balancing robustness and speed should be done on a case by case basis based on the evaluation cost of $f(t)$, $f'(t)$ and that of the intersection with the base shape and bounding box. Consider the following pathological cases:

- Several intersections: Fig. 5.12.
- Failed bracket: Fig. 5.13.

These issues can be solved by stepping through values of t carefully. This is a widespread technique in computer graphics, often referred to as *ray marching*. Many techniques exist, for instance with dynamic t step sizes depending on the proximity of the object, but this takes us outside the scope of this chapter. Note that a compromise must be found between giving up on the intersection (for the sake of speed), concluding that the ray misses the surface, accepting an intersection when we are not sure of having found the right one, and looking harder with ray marching. See also Lerner (p.66) [175] on the intersection problem, notably the concept of guide surface.

5.4 Implementation

5.4.1 Our implementation: Details and tools used

We have implemented a sequential raytracer using the core mathematical descriptions in the present chapter. Its main features are:

- 3D vectorial raytracing for off-axis systems.
- Spherical and XY polynomial surfaces (demonstrated in [144]).
- Fast: the speed of raytracing and MF evaluation is of the same order of magnitude as that of OpticStudio.
- Script User Interface (UI) to input optical systems, create MF and manage search algorithms. No Graphical User Interface (GUI).
- Ray-aiming: prototypes, more in Chapter 6.

We used Common Lisp for our implementation, with the compiler Steel Bank Common Lisp (SBCL) [176]. Read-Eval-Print Loop (REPL)-driven development has allowed us to prototype our implementation much faster that we could have in other languages, such as C++. Thanks to SBCL, which compiles Common Lisp down to machine code and provides many options to optimize low-level performance, our implementation is quite fast as we'll see.

5.4.2 Test systems and timings

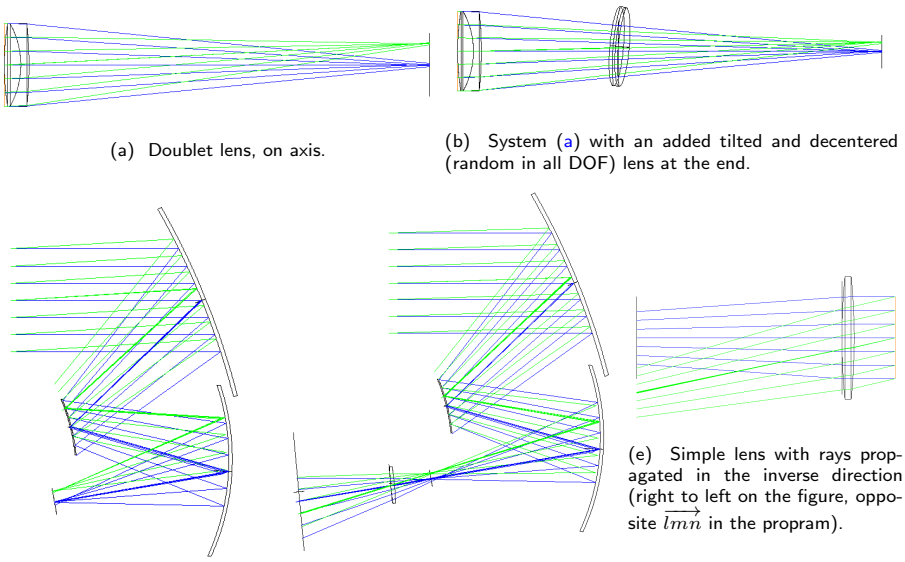
We used test systems (Fig. 5.14) to validate our implementation against Zemax (Zemax 13 SP1 EE (64-bit) Version April 4, 2013), at least in the 3D spherical case with both reflection and refraction and no ray-aiming. From a ray grid at the entrance pupil, we produced the ray data (x, y, z, l, m, n) at each surface for the two programs. We report that the data matches down to numerical machine double-float precision.

Furthermore, we benchmarked the raytracing speed of our implementation. We obtained a metric hopefully similar to the integrated Zemax benchmark function which measures a number of rays traced per second on a given optical system. In our implementation, we compute the metric by raytracing an array of a million rays through the system. The comparison is only indicative of performance since in reality, there is significant overhead for computing the MF for much fewer rays (hundreds) and treating the ray data for each MF. The results for the test systems are in Fig. 5.15. The tests were performed on the same machine on one Central Processing Unit (CPU) thread.

We notice that for systems with decentered and tilted elements (b,c and d), the raytracing speed is almost the same between both programs, with a slight advantage for Zemax[16]. On systems with completely axisymmetric components (a and e), Zemax is 2 to 3 times faster. We suspect Zemax treats axisymmetric systems as a special case and exploits the symmetry to avoid redundant computations, in contrast we always perform the full 3D computation.

Let us give orders of magnitude useful for the dimensioning of raytracing operations: Fig. 5.16.

[16]This proves, perhaps surprisingly, that Common Lisp compiled with SBCL is adequate for high-performance CPU computing. It is indeed as fast as the Zemax code, which is probably very efficient C code. With Common Lisp, we had access to the required performance where needed along with the more abstract language features which C is lacking that reduce development time.



(a) Doublet lens, on axis.

(b) System (a) with an added tilted and decentered (random in all DOF) lens at the end.

(c) Plane-symmetric TMA.

(d) System (c) with added randomly tilted and decentered lens at the end.

(e) Simple lens with rays propagated in the inverse direction (right to left on the figure, opposite $\overrightarrow{lmn}$ in the propram).

Figure 5.14: Validation and benchmark optical systems for our raytracing implementation. All diopters are spherical and raytracing starts from a known entrance pupil.

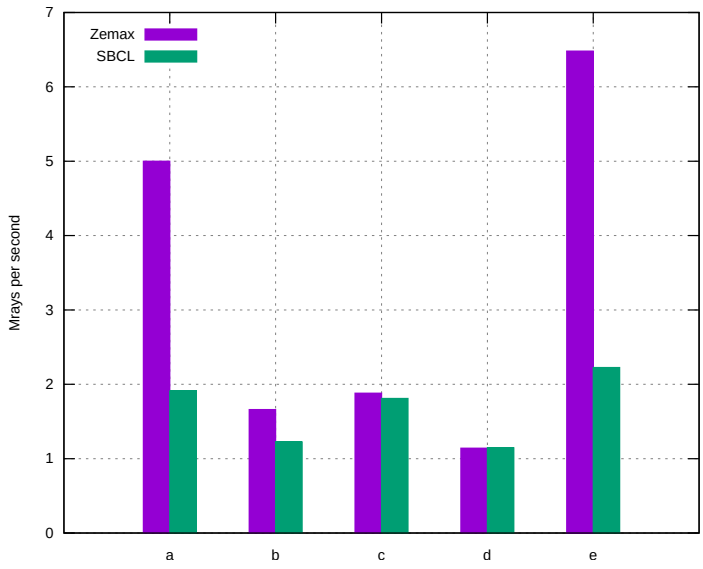


Figure 5.15: Raytracing speed comparison between our implementation and Zemax on our test systems.

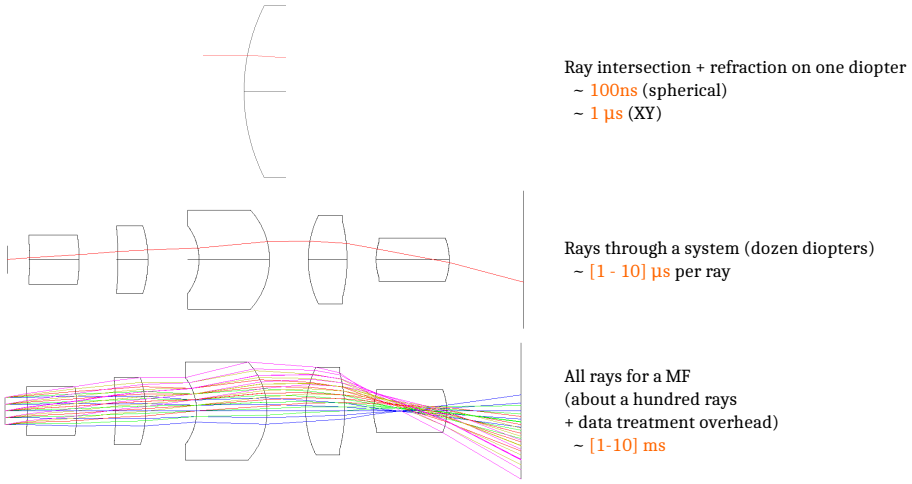


Figure 5.16: Raytracing orders of magnitude for performance in our implementation.

5.4.3 Parallelism

Raytracing is a parallel problem: first obviously at the ray level but also at the higher level of MF evaluation. Which level is actually exploitable depends on the number of rays being traced and the nature of search algorithms.

5.4.3.1 Ray-level

Each ray can be traced independently from the others through a given optical system. Indeed, sequential raytracing can be seen as the successive application of functions (intersection, refraction etc.) on each ray with an auxiliary array of parameters (radii of curvature, thicknesses etc.). This point of view is illustrated in Fig. 5.17.

We ran performance measurements of parallel raytracing at the ray level on a spherical axisymmetric Cooke Triplet without ray-aiming to assess the difference between languages and architectures (Fig. 5.18). We raytraced a million rays in each case then divided the timing to obtain an indicative measure of performance per ray[17]. We compared Common Lisp compiled with SBCL, C with a variable number of CPU threads and Compute Unified Device Architecture (CUDA) on a Graphics Processing Unit (GPU). Again, results are only indicative of real attainable performance since we trace one million rays at a time, moreover we are not expert in parallel computing or GPU computing and it is quite possible the parallel cases could be further optimised. All tests are performed on the same machine. The CPU has 4 cores and 8 threads. The rays were divided equally into

[17]The raytracing is re-implemented naively for the specific case of the Cooke Triplet so we can ignore the overhead relevant to excessive number of function calls *etc*, it is then slightly faster than general raytracing code.

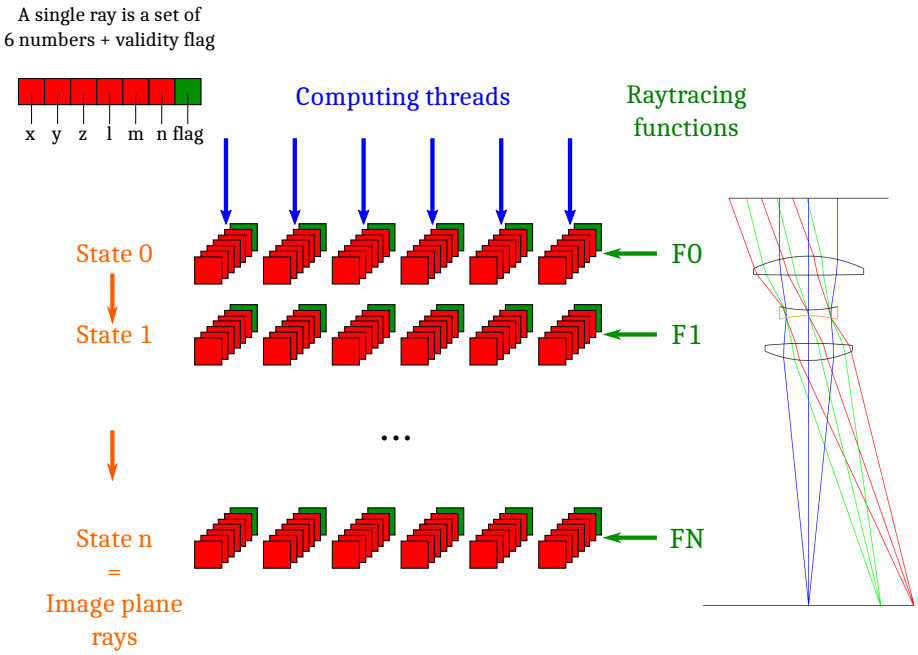


Figure 5.17: Raytracing rays in parallel through an optical system: naive view. Each computing thread applies the sequence of raytracing functions F on a ray.

each available thread. The GPU is a Nvidia GTX 1060 and the GPU raytracing is implemented in CUDA. One CUDA core per ray is used[18]. The results show that C is slightly faster that the Common Lisp implementation. Then, as we increase the number of threads involved in the computation, we see a typical (for multithreaded tasks) sublinear increase in performance. GPU raytracing is significantly faster, moreover we suspect that the results would scale almost linearly with the number of available CUDA cores on the graphics card.

This begs the question: If raytracing on parallel architecture is so fast, why isn't it used everywhere? We have to take into account that the comparison performed here is an unrealistic use-case for sequential raytracing, but much more relevant for non-sequential raytracing. We raytrace a million rays, in reality we won't have more than a few hundred rays that are required to assess the performance of optical systems. Furthermore, the raytracing of a ray is a relatively cheap operation, to the point that the threading overhead will likely be significant and completely nullify performance gains in cases where few rays need to be traced. In addition, the control flow in sequential raytracing is not so simple once ray-aiming is introduced.

In a few words, we think parallelism at the ray level is not really exploitable for typical sequential raytracing, except perhaps in cases where tens of thousands of rays need to be traced through many complex surfaces for a single MF

[18] It *might* be better to operate on several rays per core, we did not try it.

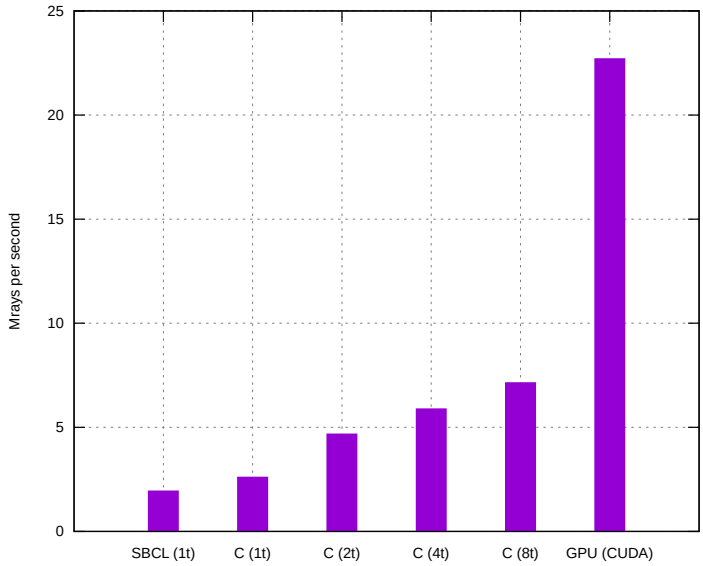


Figure 5.18: Performance timing of several naive parallel ray-level implementations on an axisymmetric spherical Cooke Triplet without ray-aiming. The number of involved threads for CPU is indicated with "nt" for n threads. SBCL: Common Lisp implementation. C: C language. GPU: CUDA language.

(for example to take into account high spatial frequencies). It is however very relevant to non-sequential raytracing, although the computations differ significantly from what we have shown.

5.4.3.2 MF level

A more adequate and flexible parallelism is at the MF evaluation level. This is what we chose in our implementation. This requires using search algorithms that allow at least some degree of parallelism. Thankfully, even search algorithms that were developed without parallelism in mind generally contain logically parallel search function evaluations:

- Population-based algorithm (*eg* Particle Swarm Optimization (PSO)) can generally allow the parallel evaluation of the MF value at the position of individuals in the search space at each step.
- Even in the simplex search, some steps like general contraction require reevaluating the MF at the position of almost all points. This can be done in parallel.

In our implementation, the PSO and Simplex algorithms are parallel. However, for these algorithms, the number of parallel MF evaluations is limited by the nature of the search method. For example in PSO, we can only evaluate in

parallel as many MF as there are particles in the swarm. Increasing the number of individuals artificially to take advantage of a parallelism is not always advisable in search algorithms that were not designed specifically for this use. This is why parallel search algorithms were developed [177, 178].

An additional, practical (but perhaps less justifiable), reason to implement parallelism at the MF level is that programming for parallel tasks is notoriously difficult and raytracing in a system that includes ray-aiming and complex surfaces is complicated enough as it is. In contrast, search algorithms are easy to implement and the complexity added by parallelism is easily managed.

5.4.4 Raytracer architecture

The mathematical ray operations given in this chapter are required but insufficient to build a usable, robust and fast sequential raytracing software. In fact, we found that most of the difficulty in implementing a sequential raytracer lies in the program architecture design. Let us outline our proposed architecture.

We may divide the program into separate subsystems[19]:

- Raytracer: The goal is to provide a *function closure*[20] that exposes the user-defined search variables to the search algorithms and which, when called, performs the raytracing of a user-defined optical system. The produced closure must be evaluable as fast as possible and in a thread-safe way.
  - OS (Optical System) definition: Transforms a user-defined OS into an OS representation suitable for fast raytracing. Among other things, the user parameters are used to pre-compute raytracing quantities (*eg* rotation matrices), to compress sequences of operations (when multiple *transfer* surfaces are adjacent and do not involve user variables, they may be compressed into just one surface in this step). The representation also contains the list of core raytracing operations to apply. The result is a closure that takes user variable values and outputs arrays of raytraced rays ready to be inputted into MF computation.
  - Core raytracing functions: These are the mathematical operations that were explained earlier in this chapter. They are functions that operate on arrays of rays and pre-computed surface parameters.
- Search: Search algorithms (see Chapter 7). The algorithms can be implemented in a completely decoupled way. Indeed even other languages or optimization libraries may be used in place of this subsystem.

[19] Or *libraries*, *packages*, whatever they are called in your particular programming language idiom.

[20] To appreciate the flexibility that closures bring to programming, we recommend the book Let Over Lambda by Doug Hoyte [179].

- MF computation & Co.: Auxiliary subsystem for computing MF values from arrays of rays. Other small, auxiliary, subsystems include a ray aperture stop sampling generator and other useful optics-related computations for the user. All these may actually be left up to the user to implement.

These subsystems are largely decoupled[21].

5.4.5 Discussion and Implementation advice

We provide some advice and opinion from our experience in implementing our raytracer.

5.4.5.1 Development focus

The mathematical operations to perform raytracing are the *easy* part. It might be tempting to develop a crude program architecture and add many surface representations (every known polynomials, Non-Uniform Rational Basis Spline (NURBS) *etc*). But we advise to only include a few surface representations and develop instead a solid program architecture. The reason is that ray-aiming and user UI could require massive changes to prototype architectures. It is better to have to modify a relatively lightweight raytracing codebase than a cumbersome ones with many features.

5.4.5.2 Modularity

The core feature of a sequential raytracer is to raytrace arrays of rays given an aperture stop sampling, an optical system, fields *etc*. Search algorithms are completely external to this core feature, the program architecture should reflect that. This allows using any third-party library to handle the search[22].

On the subject of surface representations: there are many possible types. Providing a way for the user to implement their own surfaces is useful to offload some of the development effort on the user, who may have very specific needs[23]. These user-defined surfaces must provide the rules for computing output ray data from input ray data ($xyzlmn$ in what we have presented).

The topics we have just covered, however few there were, suffice to compute the geometric propagation of rays through many, if not most, optical components (lenses, mirrors, diffraction surfaces). The specific issues of polarization, illumination quantities (etc) on the focal plane etc are left out of our work. These issues could be dealt with using the same architecture (at least as far as we reckon

[21] In particular, search algorithms and search logic implementations have nothing to do in the raytracer subsystem. They may be called for iterative intersection (1D root-finding) and ray-aiming (2D search) as general black boxes.

[22] Furthermore, if you are a programmer in optics, odds are that search algorithms libraries made by professionals in the field are better than yours.

[23] OpticStudio provides this feature as *User-defined surfaces* Dynamic-Link Library (DLL).

without having tackled the problem ourselves) but should involve tracking more ray properties through the optical systems.

If sequential raytracing were only about the contents of the past chapter, it would be rather easy. Crucially we have left out, until now, the problem of ray-aiming that is necessary to perform computations on optical systems the aperture stop of which is not at the entrance. This is what the next chapter is about.

Chapter 6

Ray-aiming strategies

Ray-aiming problem explained – Single-ray point of view – Whole sampling point of view – Naive solution for simple systems – Diving into complex issues arising from freeform and other complex systems – Dynamic aperture stop position

We find *ray-aiming* to be both a complex concept and a complex task to program in a sequential raytracing program. However, it is required to perform raytracing in the case of optical systems with an aperture stop that is not at the entrance of the system (which is rather often the case). We find the treatment of the subject in the literature to be almost non-existent[1] even though the method is obviously programmed into commercial software. As the issue is more open-ended than basic raytracing through single components, there are several ways of solving it. Rather than giving ready-made closed-form formulae, we give several lines of reasoning on how to solve ray-aiming.

We begin by framing the problem as clearly as we can, then present a naive scheme to solve the problem, which works for simple systems. Then we dive into issues arising with freeforms and unobscured telescopes.

6.1 Ray-aiming problem

Any typical optical system sequential raytracing setup possesses:

- **Fields**, provided in object space[2].
- An **aperture stop sampling** and position in the optical system.

Real raytracing requires that all the rays defined in the object space by the field direction match with the positions defined in the aperture stop sampling when raytraced from the object space to the aperture stop surface. The difficulty

[1] We apologize if we have missed something. Please let us know if perchance you have found or written *anything* on the subject.

[2] With the notable exception of fields being defined as positions on the image plane. We leave out this case in the present treatment of ray-aiming.

is that the relationship between the bundles of rays in object space and the bundles of rays at the aperture stop is governed by the laws of geometrical light ray propagation through this first half of the optical system (From object to stop). This relationship can be arbitrarily complex if numerous or complex surfaces are placed between the object space and the aperture stop. *Ray-aiming* is the process by which the rays that satisfy both the field and aperture stop constraints are found and can be subsequently raytraced normally through the second half of the system (from stop to image). Ray-aiming is not an active research area in the publicly available literature, although it was undoubtedly implemented in commercial software. We mostly used the CodeV and OpticStudio manuals on the issue (although they do not reveal complete strategies of course).

For the sake of simplicity we consider objects at infinity in our treatment of the subject. Ray-aiming problems with objects at finite distances can be reduced to problems with objects at infinity by the addition of a perfect lens as a first element in the system.

6.1.1 Optical system decomposition around the stop

We can divide any optical system into subsystems for raytracing around the aperture stop (Fig. 6.1). Four subsystems are of interest:

- **1**: The whole optical system in direct raytracing.
- **2**: The stop/object part in inverse raytracing.
- **3**: The stop/image part in direct raytracing.
- **4**: The object/stop part in direct raytracing.

For the purpose of performing ray-aiming, only subsystems 2 and 4 are needed. The subsystems 1 and 3 are used to perform the rest of the raytracing to the image plane.

6.1.2 Description for a single ray

For a given ray, we know:

- Its target position $(xyz)_{stop}$ on the aperture stop surface, as defined by the aperture stop sampling and the altitude of this surface.
- Its target direction $(lmn)_{obj}$ in the object space, as defined by the field direction.

These two quantities are constraints that the ray must satisfy, but they are insufficient to perform raytracing straight away. One must also know either a point $(xyz)_{obj}$ in object space the ray passes through or the direction $(lmn)_{stop}$ of the ray at the aperture stop surface. We use subsystems 2 and 4 in Fig. 6.1 in order to determine these quantities. The important quantities in the ray-aiming of a single ray are synthesized on Fig. 6.2a.

There are two ways to perform this search:

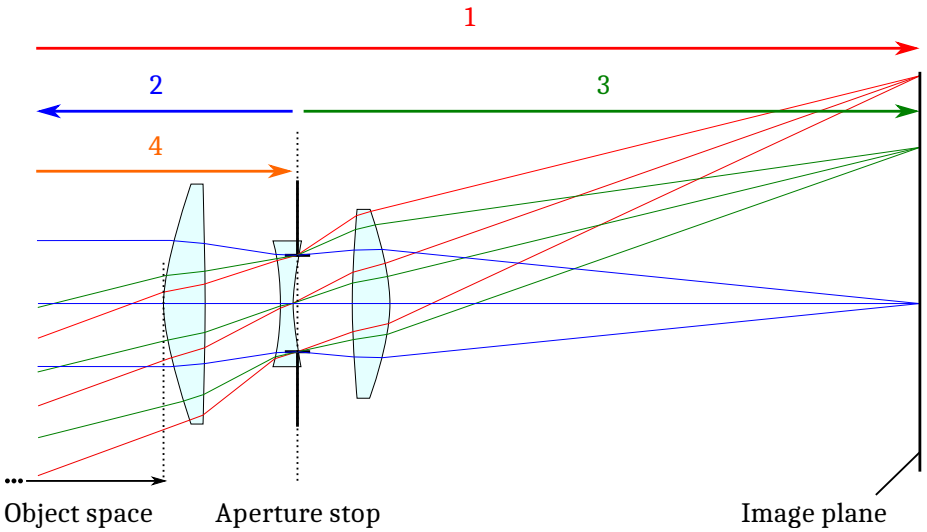
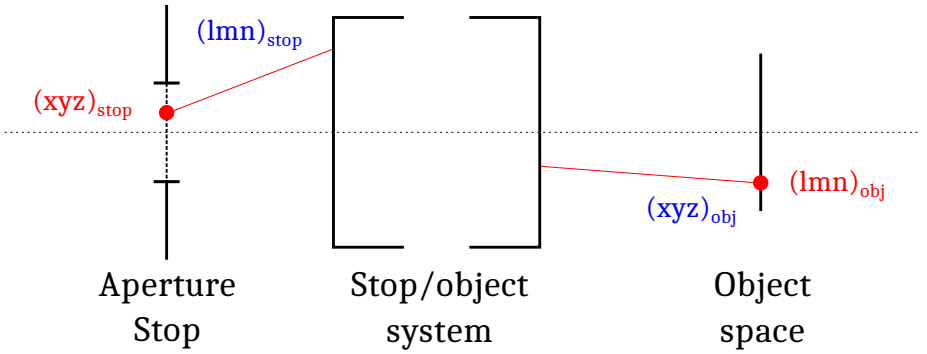
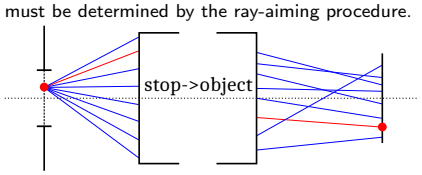


Figure 6.1: Raytracing decomposition for a typical optical system.

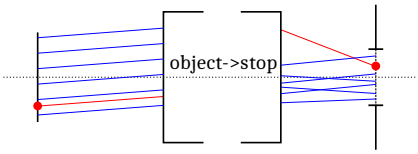
Figure 6.2: Ray-aiming for a single ray.



(a) Important ray-aiming quantities for a single ray. The quantities in red are constraints. The quantities in blue must be determined by the ray-aiming procedure.



(b) Searching from the aperture stop. The variable to optimize is the direction from the known position on the stop. The target is the direction of the raytraced ray in object space.



(c) Searching from the object space. The variable to optimize is the position in object space of a point on an arbitrary plane. The direction of the ray in object space is known. The target is the position of the raytraced ray on the aperture stop.

- Start from the aperture stop position $(xyz)_{stop}$ and search for the ray direction $(lmn)_{stop}$ that gives the correct field direction $(lmn)_{obj}$ after raytracing[3]. This uses subsystem 2. This method is illustrated in Fig. 6.2b.
- Start from the field direction $(lmn)_{obj}$ in object space and search for the ray position $(xyz)_{obj}$ that gives the correct ray position on the aperture stop $(xyz)_{stop}$ after raytracing. This uses subsystem 4. This method is illustrated in Fig. 6.2c.

Finding either $(lmn)_{stop}$ or $(xyz)_{obj}$ is sufficient to solve ray-aiming for a given ray. When one is known, the other can be found by unambiguous raytracing.

The n component of $(lmn)_{stop}$ is given by the normalization of the direction vector. Likewise, $(xyz)_{obj}$ represents only 2 degrees of freedom, since we can define our object rays at any plane in the object space. We can then describe the *ray-aiming* of one ray as a 2D search problem. We need to minimize either one of the two functions (Eqs. 6.1 and 6.2).

$$f_1(l_{stop}, m_{stop}) = d_1(rt_{stop\to obj}[ray^{stop}(l_{stop}, m_{stop})]_{lm}, (lm)_{obj,target}) \tag{6.1}$$

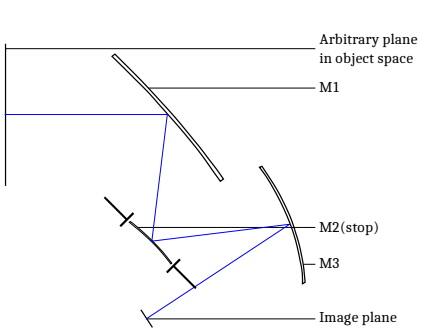
$$f_2(x_{obj}, y_{obj}) = d_2(rt_{obj\to stop}[ray^{obj}(x_{obj}, y_{obj})]_{xy}, (xy)_{stop,target}) \tag{6.2}$$

With:

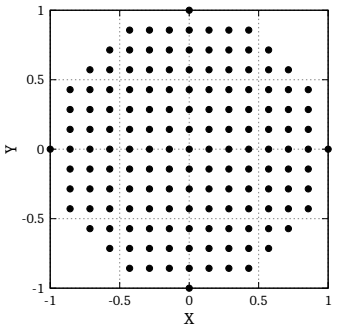
- $rt_{stop\to obj}[.], rt_{obj\to stop}[.]$ denoting the raytracing operation from the stop to the object space and object to stop respectively. We get a ray as a result of this operation and we have its xy or lm coordinates.
- ray^{stop}, ray^{obj} a ray originating from the stop and a ray originating from the object space respectively.
- d_1, d_2 arbitrary *metrics* between ray coordinates. This is the quantity to be minimized.
- $(lm)_{obj,target}, (xy)_{stop,target}$ the target coordinates for the raytraced rays in each case.

The interchangeability of these two problems can be exploited fruitfully in raytracing stategies. We can go back and forth between object and stop spaces during the search. Indeed, minimizing either one of Eq. 6.1 or Eq. 6.2 is equivalent.

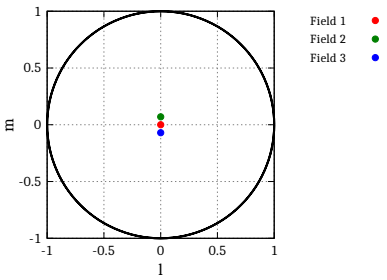
[3]This is probably where the name *ray-aiming* comes from. At a fixed position on the aperture stop, we *aim* the ray by trying several directions and looking at the result as a field direction until we find the right one.



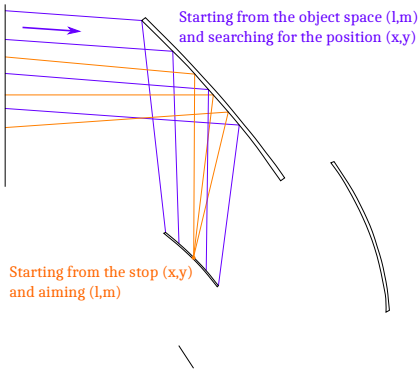
(a) Optical system surfaces definition.



(b) Aperture stop sampling scheme, here a cartesian grid with 15 rays on one side cropped to the circular stop shape. We plot normalized pupil coordinates.



(c) Defined object field directions.



(d) Search process applied on the rays of the stop samplings for each field direction. The rays involved in the search from the stop are in orange. The rays involved in the search from the object space are in purple.

Figure 6.3: Example ray-aiming process on a TMA telescope.

6.1.3 Illustrating samplings with a TMA

The problem of ray-aiming, while it involves single rays, is only solved when whole bundles of rays for a given field are correctly aimed. Let us illustrate this on a real optical system (Section 6.1.3). The example system is YATMA (Chapter 4). The ray-aiming is performed by OpticStudio in this example and the ray data was extracted for illustration purposes.

The optical system is entirely described using the optical surfaces definitions and the geometric position of the surfaces relative to each other (Fig. 6.3a). Two pieces of information are necessary to begin the ray-aiming: 1. The aperture stop sampling (Fig. 6.3b) 2. The field sampling (Fig. 6.3c). These two samplings can be seen as targets in the search problem that is the ray-aiming (Fig. 6.3d). The ray bundles, one for each field in the field sampling, must cross the aperture stop exactly at the points defined in the stop sampling and have a direction in object

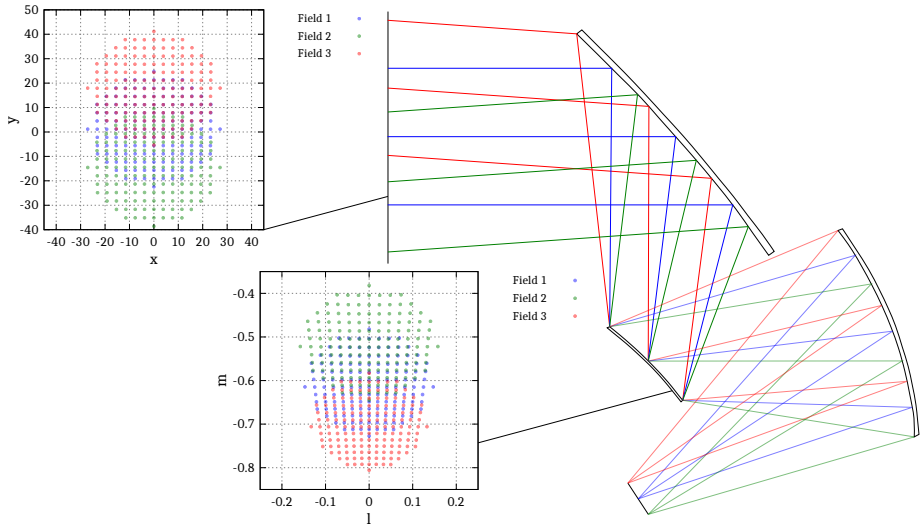


Figure 6.4: Resulting samplings in (x, y) and (l, m) ray coordinates at the stop and in object space.

space that corresponds to the relevant field. We consider the ray-aiming process to be successful when these targets are met for all the ray bundles, although additional considerations must be taken into account for complex systems (Section 6.3.4). The ray-aimed ray bundles define two complementary result samplings: 1. The position of each ray on an arbitrary surface in object space 2. The direction of each ray from the aperture stop. These two result samplings (Section 6.1.3) are complementary in the sense that, for a given optical system, one can be obtained using the other via unambiguous raytracing. It is therefore sufficient in order to give a solution to ray-aiming to provide either one of these samplings.

6.2 Naive solution

Observing that, for most systems, the ray sampling in object space is only a slightly deformed aperture stop sampling. We propose using a bilinear transform method relying on marginal rays, this is similar to what is used in paraxial raytracing. This naive solution that mostly works on axisymmetric systems. Its advantages are that it is quite fast. On the other hand, it is not very robust and can be impacted by all the problems raised in Section 6.3. It works as follows:

Given a disk aperture stop sampling of radius d_{stop}, let four marginal rays have coordinates on the aperture stop sampling at points A, B, C and D, in the (x, y) coordinate system centered on the chief ray such that:

$$\begin{cases} A = \begin{pmatrix} d_{\text{stop}} \\ 0 \end{pmatrix} \\ B = \begin{pmatrix} 0 \\ -d_{\text{stop}} \end{pmatrix} \\ C = \begin{pmatrix} -d_{\text{stop}} \\ 0 \end{pmatrix} \\ D = \begin{pmatrix} 0 \\ d_{\text{stop}} \end{pmatrix} \end{cases} \tag{6.3}$$

We can assign any point P in the sampling coordinates (u, v) such that

$$P = (1-u)(1-v)A + u(1-v)B + (1-u)vD + uvC \tag{6.4}$$

With our particular A, B, C, D coordinates, the problem is tractable:[4]

$$\begin{cases} u = \frac{1}{2} - \frac{x_P + y_P}{2d_{\text{pup}}} \\ v = \frac{1}{2} + \frac{y_P - x_P}{2d_{\text{pup}}} \end{cases} \tag{6.5}$$

Now that we have (u, v) coordinates for every point in our sampling, we proceed with ray-aiming the marginal rays individually (A,B,C,D on the aperture stop). Once this is done, we have obtained, in an arbitrary plane in object space, solution points A', B', C', D'. We simply have to apply the bilinear map for all the other points P' in the sampling to fill the entrance pupil with rays:

$$P' = (1-u)(1-v)A' + u(1-v)B' + (1-u)vD' + uvC' \tag{6.6}$$

This strategy is illustrated in Fig. 6.5. As an additional step, ray-aiming each ray individually from the candidate solution given by the above method may be required to obtain a more accurate solution for complex systems. The bilinear transform should give a candidate solution quite close to the true solution, so searches should be short.

We tested this ray-aiming strategy on the Double Gauss system in Fig. 6.6a. We validated the spot diagrams obtained using our software with our ray-aiming strategy (Fig. 6.6c) against the spot diagrams obtained using Zemax (Fig. 6.6c). Results are identical with error close to machine precision. Additional ray-aiming of every ray was not required in this case.

6.3 Problems with complex systems

Ray-aiming in complex systems, typically with strong freeform surfaces and with no clearly defined axis that a chief ray will likely follow, becomes harder.

[4] The case of general coordinates for A,B,C and D (anywhere on the (x, y) plane) allows finding (u, v) in closed form of course but the expression is longer.

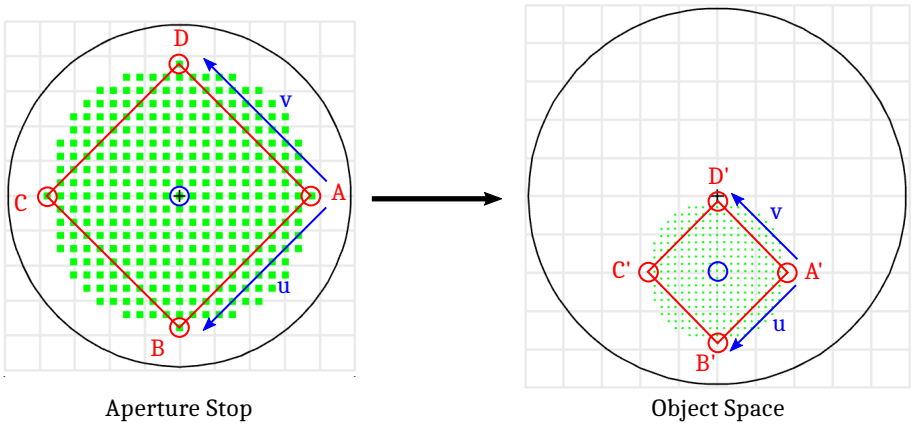


Figure 6.5: Bilinear transformation ray-aiming strategy. The aperture stop sampling (left) is transformed into a sampling on a plane in object space using the complete ray-aiming of four marginal rays and a bilinear transformation.

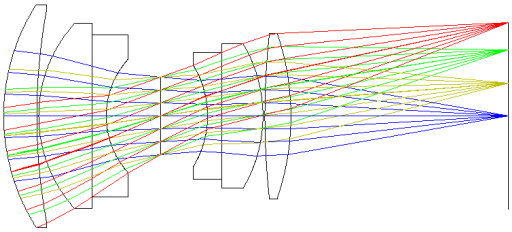
The ray-aiming search problem, thankfully, is not a black-box, we can gather information useful to the search by *opening the black box*, at the cost of increased implementation complexity. We show difficult cases that can arise in ray-aiming and some ideas indicating how to solve them. Note that some of these cases can be so exotic as to put in doubt whether or not the system is really an imager, in which case giving up ray-aiming would be perfectly acceptable (but detecting these cases is hard in itself). In the following, we mostly treat ray-aiming from the aperture stop towards the object space. Most situations can be transposed to the other ray-aiming point of view, starting from the object space.

6.3.1 Search direction and ray missing surfaces

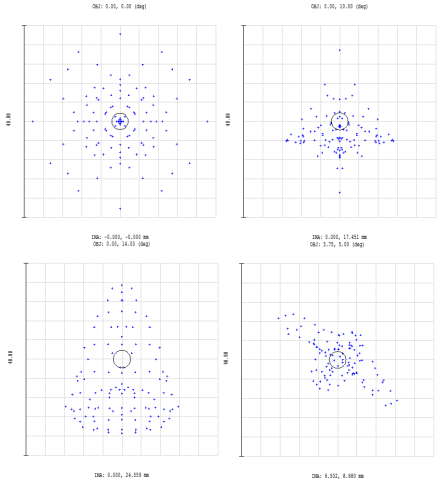
Let us say we have an off-axis system (Fig. 6.7a) and want to ray-aim a ray from point P on the aperture stop. We have M1 and M2 in front of the aperture stop, towards the object space. A problem we encounter is that the mirrors can be so small that we will likely miss them if we do not gather additional data on the system geometry to guide the search. We can, for instance:

- Start the ray parallel to the mechanical axis ($A_s A_1$).
- Start the ray oriented towards A_1.

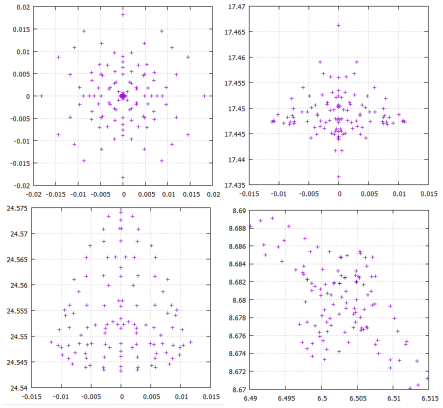
In this way, we are very likely to intersect M1. But reaching the second mirror M2 once M1 intersections have been found can be difficult (especially if the optical system does not have good imaging properties), since its direction is given by raytracing. For instance, the green ray on Fig. 6.7a reaches M1 but misses M2 completely. To solve these cases, we can think about the problem as finding the ray that successively intersects all the surfaces between the stop and the object space (Fig. 6.7b). The solution (l, m) pair is enclosed in concentric continuous



(a) System diagram and ray fans for the raytraced fields. Drawn in Zemax.

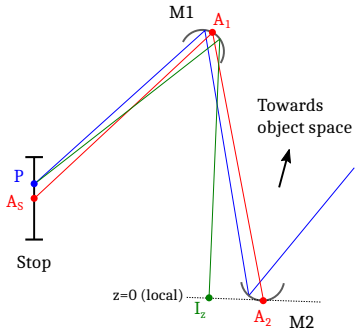


(b) Zemax raytracing results: spot diagrams at the image plane for each considered field.

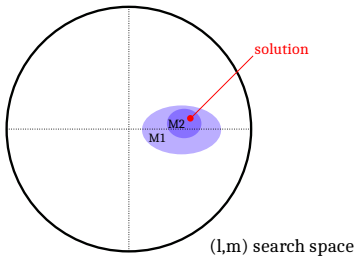


(c) Idem for our raytracing program.

Figure 6.6: Naive ray-aiming method validation on a Double-Gauss optical system. The same cartesian grid aperture stop sampling is ray-aimed and raytraced in each case.



(a) Rays are traced from point P towards the object space through successive surfaces (here mirrors). It is not known a priori whether rays will successfully intersect all surfaces.



(b) The lm 2D search space can be represented as a diagram of this shape. The areas that correspond to successful intersections are assumed to be concentric. The solution point is included in all successive sets in this case.

Figure 6.7: Finding the object space during a ray-aiming search from the aperture stop. Note we represent the intersection areas as elliptic in the lm space, but we only mean to represent continuous 2D areas within the search space.

sets of directions corresponding to whether a given mirror is intersected or not. We can encode this property in the ray-aiming scalar metric to be minimized (used when the ray direction in object space is not available to compute Eq. 6.1), for instance by a conditional function (Eq. 6.7).

$$f_{\text{sets}}\left(l_{\text{stop}}, m_{\text{stop}}\right) = \left(1 - \frac{n_{\text{inter}}}{n_{\text{total}}}\right) + \left\|\overrightarrow{I_{z,i}A_i}\right\| \tag{6.7}$$

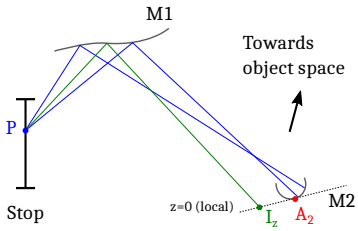
With:

- n_{inter}: Number of intersected surfaces.
- n_{total}: Total number of surfaces between aperture stop and object space.
- $I_{z,i}$: Point of intersection between ray and local $z = 0$ plane for the current surface i, when the intersection with the surface itself was not found.
- A_i: Apex of surface i.

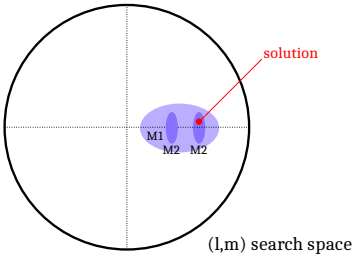
Note that we assume the ray footprint on surfaces to be somewhere close to the apex (which can be decentered by the user as needed in the surface definition of course).

The complete function to minimize for this type of ray-aiming becomes Eq. 6.8.

$$f\left(l_{\text{stop}}, m_{\text{stop}}\right) = \begin{cases} f_{\text{sets}}\left(l_{\text{stop}}, m_{\text{stop}}\right) & \text{Object space not yet reached} \\ f_1\left(l_{\text{stop}}, m_{\text{stop}}\right) & \text{Object space reached (Eq. 6.1)} \end{cases} \tag{6.8}$$



(a) M1 has a shape that, from left to right on the surface, causes rays from P to first successfully intersect M2, then miss it, and then successfully intersect it again.



(b) The shape of M1 causes disjointed areas of intersection with M2 in the search space. The solution is only in one of them however.

Figure 6.8: Ray-aiming search with strong freeform surface M1 inducing a disjointed intersection area of M2 in the lm search space.

However, with strong freeform surfaces, the problem becomes even harder (Fig. 6.8a). We very possibly can obtain disjoint areas in the lm search space for which a given surface is successfully intersected, and the solution is only in one of them (Fig. 6.8b).

A possible solving strategy is then to conduct a depth-first search in each intersection area in the search space. This would require a large number of evaluations and a criterion for concluding that a given intersection area is devoid of further intersections or solution. We did not investigate the question further.

6.3.2 Marginal rays

Using marginal rays is tempting to transform whole samplings, as shown in Section 6.2. However, in the case of freeform surfaces, these rays are likely to intersect the edge of surfaces, which likely have the strongest slopes (Zernike basis for instance). The marginal rays are therefore likely to be the first to miss surfaces in the optical system (Fig. 6.9).

This is either good news or bad news depending on the kind of raytracing we want to perform. If we want to interrupt the raytracing as soon as rays in the aperture stop sampling encounter some kind of error/special case (such as missing a surface), then this provides an early signal to interrupt the raytracing. If we want to raytrace valid rays and ignore rays with errors, then this complicates ray-aiming, as we have to design strategies that do not rely on marginal rays (or any ray in particular for that matter).

6.3.3 Problem non-convexity

Non-convexity in the search can occur, even in simple cases. On Fig. 6.10 rays are traced from P on the stop through a single mirror and their exit angle is compared with the target field direction. Going from left to right on the mirror, rays start with almost the target direction (light blue), then seem to go *away* from the target (medium blue) but finally reach the target direction at

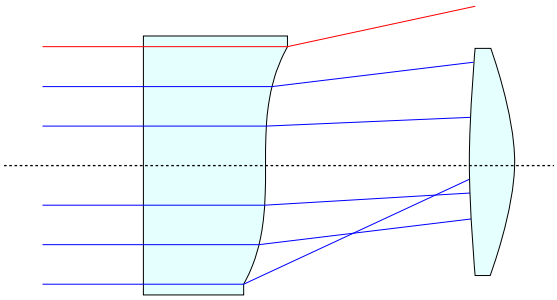


Figure 6.9: A freeform lens that makes a marginal ray (in red) go outside the optical system (miss the next surface).

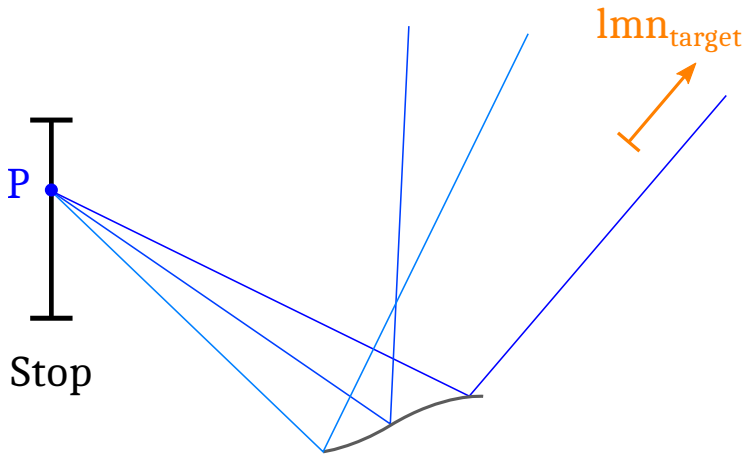


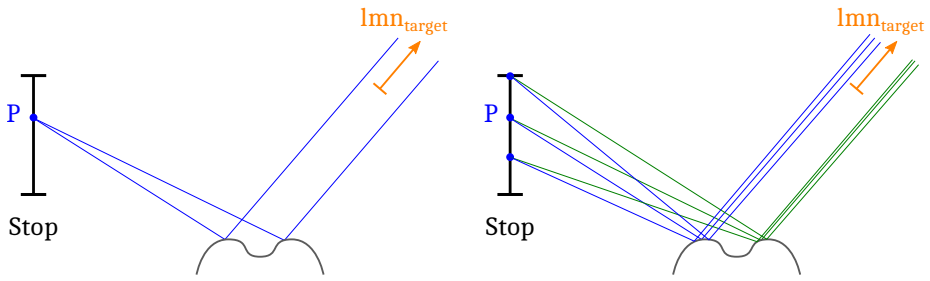
Figure 6.10: Ray-aiming problem with non-convex search properties. There is a single S-shape mirror between the stop and object space. lmn_{target} is the target field direction in object space.

the rightmost intersection (dark blue). This search landscape shape will likely defeat purely local search algorithms. This is troublesome because we generally want very light search algorithms for the ray-aiming of one ray and local search algorithms *tend* to be the lightest. Since raytracing a single ray is a very cheap operation, the search algorithm computational overhead is likely significant.

6.3.4 Solution unicity and sampling bundle

Another problem that comes up with freeform surfaces is a non-unicity in the ray-aiming solution for one ray. Fig. 6.11a illustrates this: two directions from a given point P on the aperture stop give rise to the same object field direction.

If we were to ray-aim a whole aperture stop sampling ray by ray, we could encounter a situation where rays take either possible path more or less at random. Our resulting ray-aiming, while formally a solution ray by ray, would make no sense when the whole aperture stop sampling is considered (Fig. 6.11b). Now,



(a) Ray-aiming a single ray in a case where the solution is non-unique. The ray from P on the aperture stop can follow two possible paths towards a solution in object space.

(b) Non-unicity in ray-aiming: case of stop sampling. Every ray in the stop sampling has multiple possible solutions. We have two possible solution bundles, in blue and green.

Figure 6.11: Ray-aiming solution non-unicity in the case of single rays and stop sampling bundles.

the optical system that present properties like this are almost certainly not well defined as imagers, we're almost doing something that resembles raytracing for lighting. However, we can see how one might want to have the possibility to define strong freeform surfaces that generate non-unicity and still want ray-aiming to find a single bundle. The designer could then manually cut away the part of the freeform surface that generates the spurious solution.

A possible strategy to achieve the ray-aiming of a single, sensical bundle from the aperture stop sampling might be to consider continuous functions of $(x, y)_{\text{stop}}$ into $(l, m)_{\text{stop}}$, described by general coefficients. We think rays are more likely to "stay together" using such descriptions. The search would use the coefficients as variables and raytrace all or a subset of the aperture stop sampling to evaluate the metric to minimize.

Another solution could be to search ray by ray and re-use previous results. The starting point for the search on a new ray is the search result obtained for a previous, nearby ray. This requires a way of automatically iterate through positions in the sampling while keeping a short distance between each position[5], it is straightforward if the sampling is of a known type, such as cartesian grid, polar, *etc.* Even in simple ray-aiming cases, this strategy is advisable since it is likely to speed up the search.

6.3.5 Moving the aperture stop

Optical design beginners often want to move the aperture stop automatically using search variables in sequential raytracing software, *ie* make the stop freely jump between the object and image spaces of each diopter along the system.

In our implementation, we found that it was possible to implement a feature that causes the aperture stop to be redefined in the optical system according to a variable indicating how far along the mechanical axis it should be as well as a variable for stop semi-diameter. The aperture stop is simply a flat iris when

[5] In the general case this is akin to the Traveling Salesman Problem.

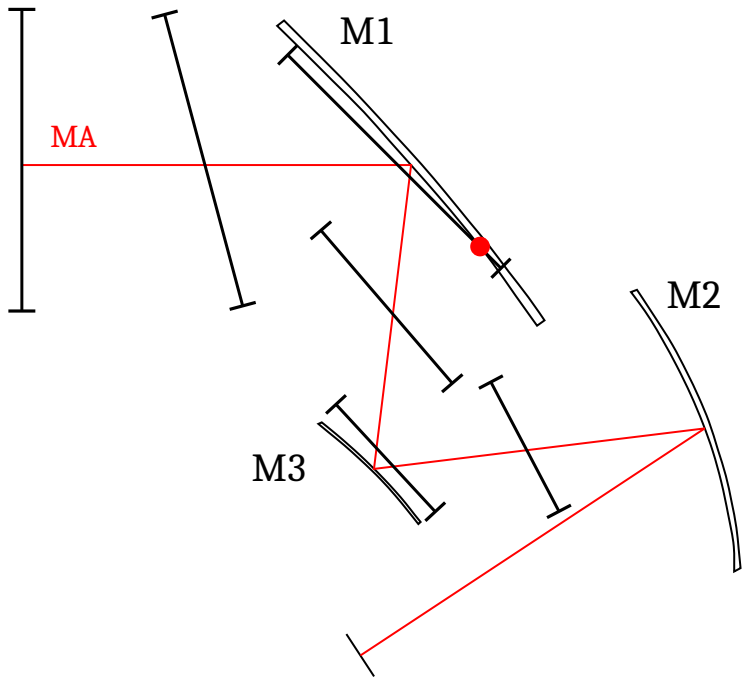


Figure 6.12: Moving the aperture stop along the mechanical axis in a TMA. MA: Mechanical Axis. The angle of the stop is interpolated between each surface. When a collision with an existing surface occurs, such as with M1 (red dot), the aperture stop can be snapped automatically to being on M1.

located in free space and snaps to surfaces whenever the iris would intersect them. In the case of off-axis systems, the aperture stop can be oriented using an angle interpolated from the *transfer* rotation matrix between the two successive surfaces. The concept is illustrated on Fig. 6.12.

We did not fully implement this feature in our raytracer yet. We think this represents a worthwhile investigation in an effort to make optical design still more palatable to search algorithms. The reasons why this concept is not commonly used could be:

- Software complexity requirements become too high.
- The strategy is not robust enough to be made generally available in raytracers.
- The search problems with such variable aperture stop become too hard.

As we have just seen, ray-aiming can be difficult in the case of complex systems. As far as our implementation is concerned, we have prototypes in various

states of completion which follow the ideas given in the past chapter. Concerning performance, even though we have not performed benchmarks on consolidated implementations, we can already say that other ideas seem to be required to get close to the execution speed reached in commercial software, which we find remarkably fast.

Now that ray-aiming is explained and solutions investigated, we have all the tools one could need to perform raytracing MF evaluation tasks on unobscured telescopes and many other types of optical systems that include lenses, mirrors and gratings. For would-be implementers out there: programming an (open-source) sequential raytracer that would include all the features we have written about in the past two chapters would already be very useful to the optical community, but only if it is also FAST (let's say realistically within 50 % *of commercial software performance). We have fallen a bit short as we have not had time to implement validated, general and fast ray-aiming schemes. Also our implementation is not likely to become public due to opposing views on intellectual property. We hope the past chapters have documented sufficiently clearly ideas that will help other programmers.*

In the next chapter, we will use our sequential raytracer (without ray-aiming) as the computational core for automatic optical design using search algorithms.

Chapter 7

Search Algorithms Applied to Optical Design

> When a lens designer studies non-spherical surfaces he is appalled by the infinite domain that opens before him.
>
> Bernard Maitenaz [3]

Levers for faster automatic optical design using search algorithms – Introduction to search algorithms – Simplex search – Nature inspired evolution strategies – Benchmarking algorithms and optimization runs – Comparing some algorithms from the literature and commercial solutions on a freeform system – Leads for further improvements in the workflow of optical design using search algorithms

The use of search algorithms on top of a sequential raytracer core is the method of choice for the automatic design of optical systems. While it is not the only approach, it is, as far as we know, the most general since it allows incorporating many objectives into the merit function. Likewise the search algorithms that are used are not specific to optical design. Since the optical design community is rather small, it is convenient that we should rely on the research effort of branches of engineering and applied mathematics with many more practitioners.

We begin by stating some of the issues that are faced when using search algorithms, as is traditional in optical design, especially in the case of designing complex freeform systems. We then give introductory ideas into search algorithms, a field which we think is sometimes obscure to optical designers but actually not hard to get into. We include some of the published works we have produced on the subject of search algorithms applied to freeform optical design, we deal with how search algorithms could be compared and how much faster existing search algorithms are than the commercial solutions in the case of a sample (by necessity limited) of optical systems. Lastly we give some of the

directions in which optical designers might go to further improve how they use search algorithms in their work.

7.1 Cost of search problems in freeform optical design

Freeform optical design has made the optimization of optical systems harder. We can cite some of the reasons:

- The number of problem parameters has increased dramatically.
- The performance analysis, including MF computation, is made more computationally expensive.
- Although this is hard to quantify and assert for all systems, it is often the case that freeform systems (off-axis in particular) MF landscapes are filled with more unevaluable regions due to TIR, rays missing surfaces or ray-aiming difficulties.

As we have seen (Chapter 2) freeform optical surfaces are represented using more parameters than spherical or conical surfaces. We have gone from representing a single diopter using one or two parameters, to using dozens of independent parameters. In addition, off-axis systems use potentially 5 position DOFs instead of just thicknesses. This creates harder search problems, which will require a higher number of MF evaluations n_{eval} to solve. A single MF evaluation has a cost MF_{cost}.

To a good approximation, ignoring the negligible cost of managing the search algorithms themselves, the required cost in order to solve a search problem P can be expressed as P_{cost} (Eq. 7.1). The transition from conventional to freeform optics increases both cost components.

$$P_{\text{cost}} = \text{MF}_{\text{cost}} \cdot n_{\text{eval}} \tag{7.1}$$

The levers of action available in order to minimize the cost of freeform search problems are:

- Reducing MF_{cost}.
  - Using more efficient aperture stop sampling strategies (Section 3.4.1) to reduce, for a given accuracy, the required number of rays used in computing some MF components (*eg* spot radius RMS).
  - Using efficient freeform surface evaluation algorithms (Appendix A).
  - Using efficient raytracing strategies (see Chapter 5).
  - Using faster computers or more adequate processing architectures (*eg* parallel computing).

- Reducing n_{eval}.
  - Using search algorithms that are more efficient for our problems.
  - Using freeform surface representations that happen to generate a more tractable search problem [67].

In the present work, most of these leads were investigated, some more in-depth than others. Let us talk in this chapter about finding more efficient search algorithms for optical design problems. This is a complementary approach to the existing efforts in comparing freeform representations in search problems [67].

7.2 An optical designer's introduction to search algorithms

Let us look at how optical design search problems relate to more general optimization problems, for which an extensive literature exists. We then give a cursory overview of how typical search algorithms work, including the *Nelder&Mead simplex search* and the family of *population-based evolution strategies*.

7.2.1 Search Problem Characteristics and Related Search Algorithms

The problem of finding minima of functions in some d-dimensional search space is very general and corresponds to an enormous corpus of literature in applied mathematics. Let us go over the characteristics of the search problems we deal with in optical design and relate them to the optimization literature in order to narrow down the kind of algorithms we should look at.

As we have seen (Section 3.3), the way in which search problems are typically set up in optical design is via a single scalar MF that synthesizes all performance metrics (*eg* MTF value) and all constraints (*eg* lens thicknesses between 0 and some value). As we'll see (Pareto Section 7.4.3), there are alternatives for setting up search problems, but let us adopt this point of view for the time being. Formally, solving an optical design search problem corresponds to finding $X^\star$ (Eq. 7.2) the vector of optimal problem parameters (freeform coefficients, thicknesses, *etc* ...), with D the search domain (usually very "unconstrained" in optical design).

$$X^\star = \underset{X \in D}{\operatorname{argmin}} \, \mathrm{MF}(X) \tag{7.2}$$

The only kind of knowledge we can gather about the MF landscape is its value at discrete points (if we omit differential raytracing we do not have access to first nor second derivatives). This is called a *black box* problem. Moreover, optical search problems are relatively *noiseless* (compared to what is considered *noisy* in the search algorithms literature, see the functions with "severe" noise

in [180]). We could potentially obtain noisy MFs if each evaluation were done using a random aperture stop sampling with few rays. This is not the case for properly setup MFs. Finally, most parameters constituting the search space are *continuous*, with the notable exception of glass refraction indices and Abbe numbers which must correspond to available glasses in manufacturers' catalogs.

To summarize, the search problems we encounter in optical design, freeform or otherwise, typically have the following general characteristics:

- Continuous, or Hybrid (when glasses are involved), also *real-parameter*.
- Low noise.
- Black box.
- Simple constraints (bounds) or even none at all (in the way optical designers typically set up the MF).
- Number of parameters in the $[10^1, 10^2]$ range.

[181] gives a nice overview of all the kinds of algorithms designed to solve the kind of problems we have just described. In addition, the BBOB (Black Box Optimization Benchmarking) competition results [182][1] and the GECCO (Genetic and Evolutionary Computation Conference) [183] provide a useful catalog of many current search algorithms and comparison methodology.

7.2.2 Demystifying search algorithms

It is our experience that optical designers often see search algorithms as mysterious, which is not surprising given that these algorithms, critical to optical design software, are bundled in commercial products with little details about their implementation. We hope to give the readers some intuition about how search algorithms operate.

7.2.2.1 Simplex Search

First let us look at the Nelder&Mead simplex search [184]. It is one of the most widespread search algorithms. It is entirely deterministic, meaning that it is composed of a fixed set of rules with no random component, always giving the same results with the same starting conditions. The simplex search involves moving a d-simplex[2] around inside the d-dimensional search space and evaluating the MF at the simplex vertices, guiding the following steps according to the evaluation results.

For the purpose of illustration, let us take the case of a 2-dimensional space (two variables), which involves a 2-simplex (a triangle). All the search steps can be summarized on a flowchart (Fig. 7.1). At the start of the algorithm,

[1] Also `https://coco.gforge.inria.fr/` for up-to-date results.

[2] A d-simplex is a simplex in d-dimensional space. Triangles are 2-simplices, tetrahedra 3-simplices *etc.*

the three points are: best (so far) MF value position (1), second worst (2) and worst (3) (In higher dimensions, there would be other points between 1 and 2). We then apply geometric transformations on the simplex and update the points according to the decision tree. Once an end state has been reached, we loop back to the beginning. We do so until the simplex has reached a stable state, as checked by a termination criterion, either because the vertices are all superposed onto a local minimum, or because there was an early convergence due to some issue like simplex degeneracy (the vertices are aligned). Further details were explained in [144], and in the specialized literature of course. We hope the reader will appreciate that nothing mysterious is going on, as these simple steps can be followed with pen and paper on a printed-out 2D heatmap for example. Furthermore, it can be said about the simplex search, in contrast with some other algorithms, that it is *local* in nature. Loosely speaking, in 2-space, if the three vertices start in a particular "valley", there is a good chance the simplex will converge at the bottom of this valley (or become stuck due to some arrangement of bumps along the way).

7.2.2.2 Nature-inspired population-based evolution strategies

A broad and common family of search algorithms is the *population-based* and *nature-inspired* one. We illustrate some of the concepts used in these algorithms on Fig. 7.2.

- A *population* (also *swarm*) is composed of *agents* (also *individuals*) that are points in the search space at which the MF value is evaluated.
- The position of agents is updated from one step to the next using previous results, a set of rules, potentially pseudo-random components and in some cases using a velocity or acceleration vector in the search space.
- Update rules can depend on the past evaluations at a particular agent's successive positions. These rules, independent of the other agents, are said to be *cognitive* (*i.e.*, the agent is intelligent on its own).
- Update rules can also depend on the results of other agents. These rules are said to be *social*, (*i.e.*, the behaviour of an agent is influenced by other agents in the population).
- In some strategies, agents are grouped by *colonies*, *ie* a subgroup in the population that is more closely connected by rules. Agents generally interact within their colony and colonies can interact with each other.

Lastly, the name of these strategies often refers to the behaviour of various animals (fireflies, cuckoos, elephants ...). These nicknames are largely used as mnemonic devices for the rules governing the agents, the actual relation between real-life behaviour and the virtual "animal" agents is very loose.[3]

[3] A somewhat tongue-in-cheek list of such names for search algorithms can be found at https://github.com/fcampelo/EC-Bestiary

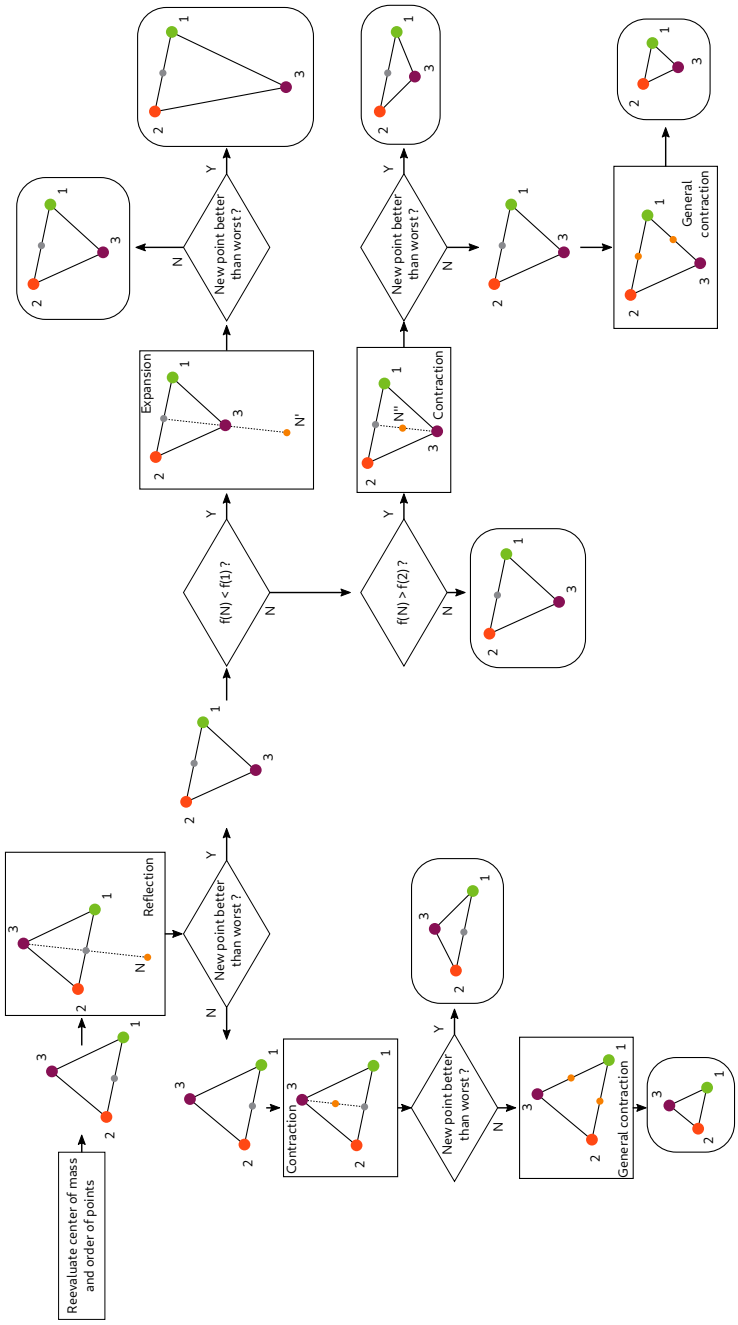


Figure 7.1: Flowchart of the simplex search applied to 2-space (inspired by the flowchart Fig 5 in [185]). The vertices of the simplex are numbered 1, 2, 3 and each a different color. Potential update vertices are smaller and in light orange.

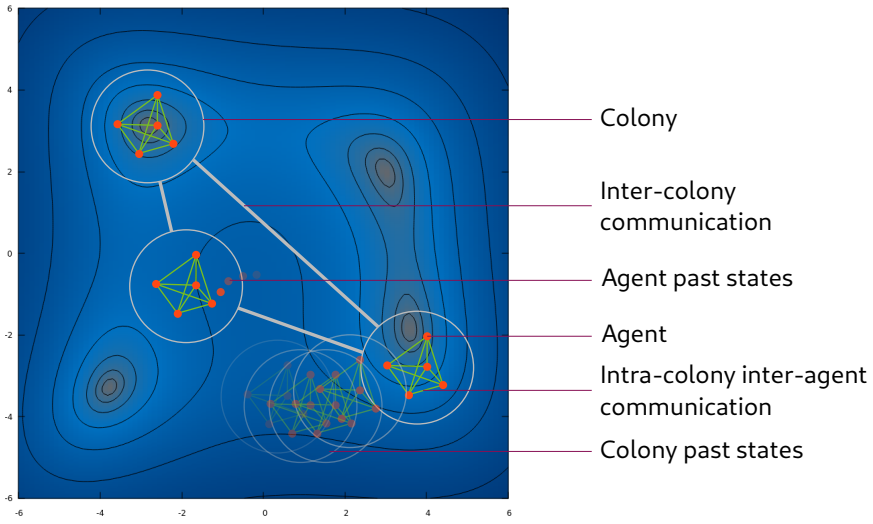


Figure 7.2: Illustration of the usual concepts in population-based nature-inspired search algorithms. [186]

To further give the reader a grasp on search algorithms, we provide URLs to animations of 5 of these algorithms operating on the 2D Himmelblau function [187]:

- Nelder&Mead simplex search [188].
- Gravity Search Algorithm [189].
- Particle Swarm Optimization [190].
- Covariance Matrix Adaptation Evolution Strategy [191].
- Cuckoo Search [192].

We note that the strategies present visually very different behaviours, which we wouldn't have noticed if it weren't for visualizations. Visualizing is also a good, convincing validation for our implementations of these algorithms (at least in 2D).

7.3 Our contributions on search algorithms applied to optical design

We [144] have compared 5 search algorithms on two eyepieces: one freeform and the other conventional. One of the point of interest is introducing a more rigorous method for algorithm benchmarking, taken from the specialized literature and

applied to optical design. It is our opinion that articles dealing specifically with the optimization of optical systems should rely more on benchmarking methods that give information about the behaviour of an algorithm on a specific problem. The usual reporting of a best MF value after some number of evaluations is insufficient[4].

We also worked on more advanced search algorithms with a researcher in the field [193], buiding on initial results using the algorithms we had presented in [186]. The optical system is a simple Cooke Triplet and the MF a penalization of Seidel aberrations. We have shown that much better results[5] were obtained using the more advanced CoMEC (Mind Evolutionary Computation co-algorithm) despite the apparent simplicity of the problem. We hope this gap in algorithm efficiency could translate to freeform optical design but we have not tested it.

We have explained some of the present context and ideas in past presentations [186, 194].

7.3.1 Notes on working with existing tools and across disciplines

There are some limits that affect the field of optical system optimization. First we often have to rely on existing commercial raytracers for complex systems, which provide (perhaps understandably) limited information about their search routines. Interfacing with these tools in order to use them solely for raytracing and be able to research algorithms in this way is possible, but entails some overhead and requires specialized knowledge.

In our work, to have superior control, we have chosen to rely on our own raytracer. But the difficulty of ray-aiming (see Chapter 6) prevented us, at the time, from looking at systems with the aperture stop away from the first surface.

Interdisciplinary work, between optical designers and optimization researchers, is also hampered by cultural differences. The optimization community (and to an even greater extent the encompassing machine-learning community) has recognized sooner the value of open-source for research, while we find this culture to be almost non-existent in optical design circles. We can however set up interdisciplinary experiments using compiled, black-box pieces of software. This is what we have relied on in our interdisciplinary work. It involves a work overhead and adds confusion though.

We reproduce hereafter the article [144] which summarizes the core of our contribution.

[4] For example, simply changing the evaluation budget and giving only an average MF at the end will often lead to some algorithm being ahead of others, but it could end up last were the evaluation budget to be doubled. This explains partly why papers on comparing search algorithms will often conclude that whatever the proposed algorithm is outperforms everything. This is a strong bias that is not necessarily conscious and that is partly remedied by the benchmarking tools developed by experts in search algorithms.

[5] At least mathematically. Since we only modeled third order Seidel aberrations, past some low threshold in MF, the performance should be estimated using raytracing instead in order to produce more realistic designs.

Research Article | Vol. 27, No. 13 | 24 Jun 2019 | OPTICS EXPRESS 18940
Optics EXPRESS

Comparing optimization algorithms for conventional and freeform optical design

THOMAS HOULLIER[1,2,*] AND THIERRY LÉPINE[1,3]

[1]*Univ-Lyon, Laboratoire Hubert Curien, UMR CNRS 5516, 18 rue Benoît Lauras, 42000 Saint-Etienne, France*
[2]*Sophia Engineering, 5 Rue Soutrane, 06560 Sophia Antipolis, France*
[3]*Institut d'Optique Graduate School, 18 rue Benoît Lauras, 42000 Saint-Etienne, France*
**thomas.houllier@sophiaengineering.com*

Abstract: Five search algorithms from the literature of black-box optimization were implemented and applied to optical design problems. These algorithms are the Particle Swarm Optimization, Gravity Search Algorithm, Cuckoo Search, Covariance Matrix Adaptation Evolution Strategy and Nelder&Mead Simplex search. The performance of these search algorithms' implementations was assessed using the BBOB2009 (Black Box Optimization Benchmark) benchmark suite. These algorithms were compared in the context of two optical case studies, one with conventional rotationally symmetric optics and one with freeform optics. A comparison was performed against a commercial optical design software. Finally we provided a simple restart scheme applicable in the workflow of an optical designer. To the best of our knowledge, this is the first in-depth quantitative comparison of optimization algorithms for optical design.

1. Introduction

Search algorithms play a crucial part in optical systems design and the literature shows longstanding efforts in the discipline [1–7]. In addition, the field of freeform optics, which commonly designates the study of optical systems with optical surfaces that do not have an axis of rotational symmetry, allows the design of very compact, very performant systems [8–12]. The use of freeform surfaces creates, as a general rule, optimization problems with more DOF (degrees of freedom) than in conventional optics. This makes the optimization process more difficult. The problem has prompted an interest in investigating various mathematical representations for freeform surfaces in their interplay with the search process [13–16]. We felt however that there was a need to compare quantitatively different search algorithms when applied to optical design examples, both conventional and freeform. We were encouraged by the good results of Menke's research in applying Particle Swarm Optimization to optical design [17]. An investigation similar to our own, limited to conventional optics and with less detailed quantitative assessments can be found in [18].

Our aim is mainly to give optical designers an insight into what they could gain from using different search algorithms on their optimization problems. We would also like to show search algorithms specialists how their field is applied to optics, so that hopefully they will be inspired to make more recent, more performant advances available to the optical design community. In contrast with typical optical design methods where a starting point is required, we investigate algorithms which only need a defined search space. This brings the optical systems optimization closer to a fully automated process.

Other research, complementary to our own in the attempt to better the optimization of freeform optics, can be found in [19–22]. These provide much needed analytical and systematic methods for the design of freeform optics and focus on providing good starting points. In contrast, we take the more common raw raytracing approach of optical design

#367023 https://doi.org/10.1364/OE.27.018940
Journal © 2019 Received 8 May 2019; revised 5 Jun 2019; accepted 8 Jun 2019; published 20 Jun 2019

Optics EXPRESS

software and focus on a more automatic optimization of optical systems by the use of various search algorithms.

In Section 2, we give an overview of where our work is situated in a typical optical design workflow. In Section 3, we describe the search algorithms we implemented using resources from the "Derivative-free optimization" and "Black-box optimization" literature. In Section 4, we then assessed their performance on a benchmark set from the literature. In Section 5, the performance of these algorithms was measured on two optical systems and compared with the performance of the search methods in the commercial optical design program OpticStudio. For this purpose, an independent raytracing tool was developed. Finally we give in Section 6 a practical template for a basic restart scheme and apply it to one of the optical systems.

2. Optical design workflow

A typical optical design project, like designing a photographic lens for example, can often be decomposed from the optical designer's point of view in the way drawn in Fig. 1.

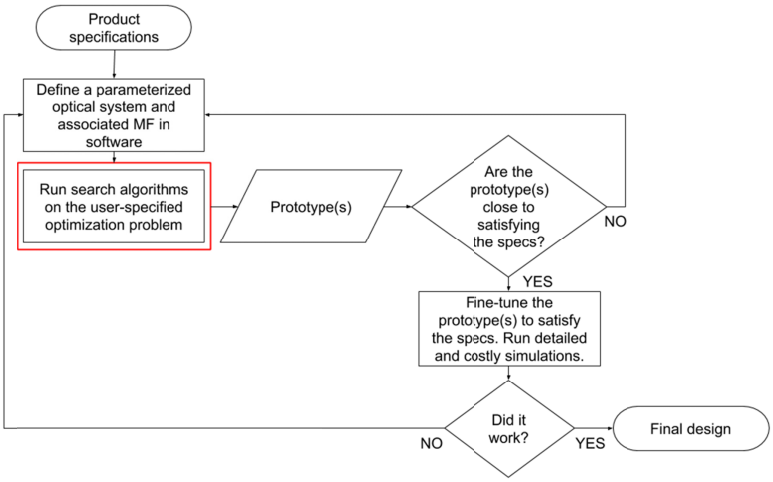


Fig. 1. Typical flowchart for the work of an optical designer. The process highlighted in red is the one we focus on in the present paper.

The MF (Merit Function) corresponds to a scalar performance score that the optical designer defines. The MF is a function of a vector of the optical system parameters. These parameters can be radii of curvature of lenses, lens thicknesses, geometrical positions of elements in the system, glass refraction indices, etc. The optical designer builds the MF to take into account basic specifications like the focal length of the optical system, the F-number, image quality etc. Advanced specifications like manufacturing tolerances or low levels of stray light are not commonly included in this "prototyping" MF, as they are too computationally costly. Each MF evaluation in the "prototyping" phase typically requires a raytracing simulation through the optical system of a few hundred rays.

Minimizing the MF means finding the set of parameters that gives the best performance according to the defined MF. Optical design software almost always includes an integrated optimization engine, often with different search methods. These search methods are not publicly disclosed in full details, at least not in commercial software.

We believe the prototyping phase to be crucial in the workflow of optical design work. In the present paper, we focus on the search algorithms used to minimize the MF.

Optics EXPRESS

3. Implemented search algorithms

We implemented 5 search algorithms from the derivative-free optimization literature. We also implemented a simple Monte Carlo search (abbrev. MC), which serves as a baseline for "worst" search algorithm. Our implementations may deviate from the canonical versions of each of the search algorithms, as numerous variations for each algorithm already exist in the literature.

Each algorithm has a set of tunable parameters, whose values can have a very significant impact on performance and must be adapted to the particular search problem the algorithm runs on. We describe each algorithm only sufficiently for the readers to understand general ideas and where the parameters come into play in our implementations. For the sake of conciseness, we refer the readers to the articles given as references for complete definitions of each algorithm.

3.1. Particle Swarm Optimization

The Particle Swarm Optimization (PSO) involves a swarm of particles, each having a position in the search space and a corresponding function value [23–25]. Each particle remembers the best position it has personally visited (cognitive component). It also knows the best position currently occupied by an individual of the swarm (social component). We implemented the so-called *gbest* swarm topology [23], where the social component is constituted only by the current best particle in the swarm. We note that PSO has already been applied to optical design with success [17].

The speed of each particle is updated at each iteration step in a loop by the formula [25]:

$$\overrightarrow{v_{next}} = \alpha \times \overrightarrow{v_{last}} + \beta \times \left(rand_1 \times \overrightarrow{D_{cog}} + rand_2 \times \overrightarrow{D_{soc}} \right) \tag{1}$$

- $\overrightarrow{v_{next}}$: The updated velocity vector of the particle.
- $\overrightarrow{v_{last}}$: The velocity vector of the particle from the previous iteration. Also called inertia.
- $rand_1, rand_2$: Two different scalar random numbers from a uniform random number generator, from 0 to 1.
- $\overrightarrow{D_{cog}}$: The distance vector between the particle and the best position it has personally visited.
- $\overrightarrow{D_{soc}}$: The distance vector between the particle and the best particle in the swarm.
- α, β : Two parameters for the algorithm. The recommended values when no information is available on the function to be minimized are $\alpha = 0.7298$ and $\beta = 1.4961$ [25].

Another parameter for the PSO is of course the number of particles we choose to use, which we will call n_{parts} .

3.2. Gravity Search Algorithm

The Gravity Search Algorithm (GSA) [26,27], like the PSO, is a particle-based search method. Each particle is seen as a planet in an n-body gravitational system. A mass is assigned to each particle in relation to the function value at its position. To make the algorithm better suited to minimization, only the heaviest planets (with better function value) exert an attraction pull. The number of planets in this active set is $K_{best}(t)$ out of n_{parts} planets in total. Furthermore, the size of the "heaviest planets" set decreases linearly with time, with only one planet attracting all the others by the end of the optimization run. The gravitational constant $G(t)$ of this fictitious universe is also dynamically decreased as a function of current iteration number.

Optics EXPRESS

We use the following update equations, as a function of t the iteration number, with n_{iter} the total number of iterations in a given optimization run:

$$G(t) = G_0 \exp\left(-\frac{\alpha t}{n_{iter}}\right) \tag{2}$$

$$K_{best}(t) = round\left(n_{parts} - \frac{t\left(n_{parts}-1\right)}{n_{iter}}\right) \tag{3}$$

G_0 and α are parameters for our implementation of GSA. $K_{best}(t)$ decreases linearly with t from n_{parts} to 1. The rest of the implementation follows closely the one outlined in [26].

3.3. Cuckoo Search

The Cuckoo Search (CS) involves particles viewed as cuckoos [28–31], who jump from position to position in the search space using a random walk modeled by Levy flights and creating 'nests' in the process at the new positions. These nests have in turn a chance of being discovered by a virtual host bird with a fixed probability (thus mimicking the parasitic brooding behavior of cuckoos). This prompts another, different, random walk using a uniform probability density. We followed closely the MatLab implementation provided in [30].

For simplicity's sake, let us consider each cuckoo as a particle that can probe around its position using two random walks: one being global, a Levy flight, applied to every particle, and the other being local and involving only a fraction of the particles. The position of a particle is updated only if it finds a better position while probing.

The position $\overrightarrow{x_{probe,Lévy}}$ probed by a particle by (global) Levy flight from position $\vec{x}$ is:

$$\overrightarrow{x_{probe,Lévy}} = \vec{x} + 0.01\sigma N(0,1)\left|N(0,1)\right|^{-0.66} N(0,1)\left(\vec{x} - \overrightarrow{x_{best}}\right) \tag{4}$$

And the position $\overrightarrow{x_{probe,walk}}$ probed by a particle via local random walk, with probability p_a of happening is:

$$\overrightarrow{x_{probe,walk}} = \vec{x} + rand \times \left(\overrightarrow{x_{shuff1}} - \overrightarrow{x_{shuff2}}\right) \tag{5}$$

- σ : A fixed parameter for Levy flight. $\sigma = 0.6965745$.
- $N(0,1)$: Random number drawn from a normal distribution with zero mean and unit standard deviation. Each occurrence in the equations is a different random number.
- $rand$: Random number from a uniform distribution between 0 and 1.
- $\overrightarrow{x_{best}}$: Position of the current best particle in the swarm.
- $\overrightarrow{x_{shuff1}}, \overrightarrow{x_{shuff2}}$: Positions of two particles in the swarm taken at random.

The parameters for this particular implementation of CS are then n_{parts} (the number of particles to use) and p_a.

3.4. Covariance Matrix Adaptation Evolution Strategy

The Covariance Matrix Adaptation Evolution Strategy (CMAES) is a population-based search algorithm [32–35]. The general idea behind this algorithm is to draw generations of points at random positions using a probability density model. The probability density, which is a multivariate normal distribution, characterized by a mean and a covariance matrix, is recalibrated every few generations to reflect the results obtained by the previous generations of points and guide the population towards a minimum. The technical details of

Optics EXPRESS

implementation are too involved to be explained here in a concise way, but the reader can refer to a very detailed guide published by Hansen on his algorithm [33], which we tried to follow as closely as possible. We use all the default parameters detailed in [33], leaving only σ, the initial step-size for mutation, as a control knob for the user.

3.5. Nelder&Mead simplex search

The simplex search developed by Nelder and Mead (SPX) is one of the most common and better known search algorithm [36–38]. It is not stochastic, meaning that it has no random component and will always give the same result with a fixed set of starting conditions. The method involves a simplex, a geometrical shape which has $D+1$ vertices, D being the dimensionality of the search space. The simplex search is based on a simple set of rules involving the function values at the positions of three points: the best, second worst and worst points of the simplex. A flowchart can be found in [36]. These rules give rise to four different possible geometric transformations of the simplex called reflection, expansion, contraction and multiple contraction (or general contraction or shrinking). The first three are homothetic transformations of the worst point in the simplex around the barycenter (non-weighted) of the rest of the points in the simplex. The new position $\overrightarrow{x_{new}}$ is obtained from the worst point position $\overrightarrow{x_{worst}}$ and barycenter position $\overrightarrow{x_{bary}}$ via:

$$\overrightarrow{x_{new}} = \overrightarrow{x_{bary}} + \alpha \times \left(\overrightarrow{x_{bary}} - \overrightarrow{x_{worst}} \right) \tag{6}$$

The multiple contraction transformation shrinks all the points (here each noted $\overrightarrow{x_{last}}$) in the simplex towards the best point $\overrightarrow{x_{best}}$ by the ratio β as such:

$$\overrightarrow{x_{new}} = \overrightarrow{x_{best}} + \beta \times \left(\overrightarrow{x_{last}} - \overrightarrow{x_{best}} \right) \tag{7}$$

Our implementation follows closely the one from Numerical Recipes [38], and we kept the canonical transformation parameters for reflection, expansion and contraction of $\alpha = -1, 2, 0.5$ respectively and $\beta = 0.5$. Our implementation of simplex search without restart strategy has no user parameter.

3.6. Other implementation concerns

Comments must be added on three issues, namely search starting point, search space bounds management and restart strategies.

Our search methods do not start at a given starting point defined by the user, as is usually the case in optical design software. The user only provides a minimum and maximum value for each variable, defining a search space boundary. We initialize every search method by drawing random points in the search space within the user-provided bounds, until we gather a sufficient number of feasible points to start the search. Feasible points are positions where the merit function can be evaluated. It is often the case in optical design that merit functions have unevaluable regions (take the case of rays missing elements entirely for example). For PSO, GSA and CS, we draw all the particles in the initial swarm at random, they are thus spread all over the search space. For CMAES and SPX, we draw a single initial random point and initialize the other particles at starting positions close to the initial point using a small fraction of the search space size in each dimension to spread apart the points while keeping them relatively close to the initial point.

The algorithms we present here generally do not have bound management methods in their canonical version. We implemented strict bound enforcing for each algorithm, we never evaluate the merit function outside of the user-provided bounds and set the eventual infringing points at the nearest bound instead. For CMAES, we used another method, we strongly penalized points falling outside of bounds by assigning them a huge constant value

Optics EXPRESS

instead of evaluating the merit function. This is not ideal, but it is simple. A more thought out way of implementing bound enforcement for CMAES is discussed in [33], it was too involved for us to implement easily.

An "evaluation budget" is a maximum number of function evaluations a search method is allowed to perform before stopping. For simplicity's sake and for ease of comparison, we did not provide any stop condition to our search methods other than the function evaluation budget. To highlight the effectiveness of restart strategies, we do however show throughout this paper the effect of adding a "stop and restart" condition to the simplex search. We designate by "restart strategies" the methods one can use to abort and restart an optimization run before having spent the whole evaluation budget, when an early convergence is detected for example. The remaining budget is used to find potentially better minima elsewhere in the search space. This stop condition, which is the one used in [38], is as follows:

$$\frac{2\left|y_{hi}-y_{lo}\right|}{\left|y_{hi}\right|+\left|y_{lo}\right|+\varepsilon} < f_{tol} \tag{8}$$

- y_{hi} : Value of the merit function at the highest (worst) point in the simplex.
- y_{lo} : Value of the merit function at the lowest (best) point in the simplex.
- ε : Tiny numerical constant to avoid division by zero.
- f_{tol} : Stop condition parameter. Note that it is independent of actual merit function value.

If the stop condition is reached, the search method aborts the current simplex run, saving the best result it had reached beforehand, and reinitializes a simplex randomly within the search space. This continues until the search method has spent its function evaluation budget.

4. Results on the BBOB2009 test suite

Given that we implemented our own versions of the search algorithms cited in the present paper, we needed to provide an objective point of reference for their performance. We chose to run the BBOB2009 (Black-Box Optimization Benchmarking) [39], which we re-implemented using the very useful online resources provided by the authors of this test suite [40].

The BBOB2009, consists of 24 different function types of D dimensions [41], which have a search space of $[-5;5]^D$, a random minimum position $\overrightarrow{x_{opt}}$, and a random minimum value of $f_{opt} \in [-1000;1000]$ drawn from a Cauchy distribution. We draw 15 optimization problems for each function type and for each dimensionality D considered, $D=[2,3,5,10,20,40]$, thus creating $24\times15\times6=2160$ independent optimization problems for our search algorithms to run on.

We run all the implemented search algorithms on each of the 2160 optimization problems and we save the best found minimum so far f_{best} as a function of number of function evaluations for each problem. For the optimization problem i having a global minimum $f_{opt,i}$, we compute $\Delta f_i = f_{best,i} - f_{opt,i}$. The optimization problem is considered as solved when Δf_i is below an arbitrary threshold Δf_{thresh}. A measure of performance for a search algorithm is then the proportion $R \in [0;1]$ of the 2160 problems which have reached the threshold as a function of two parameters, first the number of function evaluations divided by D the dimensionality of the problem $n_{eval,D} = \frac{n_{eval}}{D}$ and secondly the threshold Δf_{thresh}. This performance metric can be thought of as a 2D surface, the altitude of the surface being the proportion R of solved problems as a function of the two parameters $n_{eval,D}$ and Δf_{thresh}.

Optics EXPRESS

The results are synthesized in a way very similar to that adopted by the authors of the BBOB2009 [42], namely we draw two graphs, which are cuts of the 2D surface mentioned above:

1. Proportion of solved problems as a function of number of function evaluations per search space dimension with a fixed threshold distance to the global minimum value. This is in Fig. 2.
2. Proportion of solved problems as a function of threshold distance to the global minimum value at a fixed number of function evaluations. This is in Fig. 3.

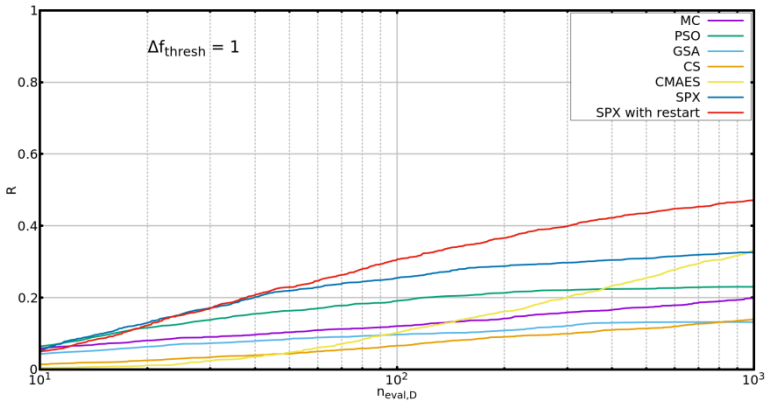


Fig. 2. Result of the implemented algorithms on the BBOB2009 test suite. Proportion of solved problems as a function of function evaluation budget. The threshold of the distance to the global minimum for a problem to be considered as solved is chosen arbitrarily to be Δfthresh = 1.

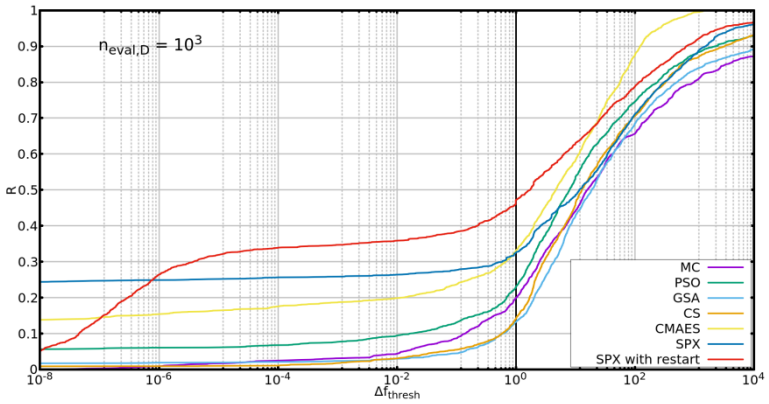


Fig. 3. Result of the implemented algorithms on the BBOB2009 test suite. Proportion of solved problems as a function of threshold criterion Δfthresh. The function evaluation budget is 10^3 per problem dimension for every algorithm. The thick vertical line highlights the Δfthresh value taken in Fig. 2. The proportion of solved problems for each algorithm at this threshold matches the proportions obtained after 10^3 function evaluations per problem dimension in Fig. 2.

A good algorithm on Fig. 2 will have a curve that is towards higher R values (more problems solved) as early as possible in terms of evaluation count $n_{eval,D}$. On Fig. 3, a good algorithm will exhibit a curve closer to the top left of the diagram, meaning that it will solve many problems with a solution closer to the global minimum. The Δf_{thresh} value can be seen as a difficulty setting meaning "solve the problems to within Δf_{thresh} of the global minimum".

Optics EXPRESS

The parameterization for all the algorithms we ran on the BBOB2009 can be found in Table 1. It is important for the readers to note that we do not pretend to compare in absolute terms the merits of the search algorithms we chose to implement, we merely state the performance of our own implementations, which could be faulty, with a particular set of search parameters, which can be ill-chosen for a given problem.

Our results show that our implementations of search algorithms do not perform very well against state of the art methods [42], which is not surprising given that we used very bare versions, with little attention to parameter tuning and no restart strategies. Despite the poor performance of our implementations against the state of the art, we will see that they nonetheless perform satisfactorily in optimizing optical systems when compared with a commercially available tool.

Table 1. Parameterization for each of the algorithms running on the BBOB2009 test suite.

Algorithm	Parameters used
PSO	$\alpha = 0.7298; \beta = 1.4961; n_{parts} = D$
GSA	$G_0 = 10; \alpha = 20; n_{parts} = D$
CS	$n_{parts} = D; p_a = 0.75$
CMAES	$\sigma = 0.1$
SPX	-
SPX with restart	$f_{tol} = 10^{-8}$

5. Application to two optical systems

We measure the performance of our implementations of search algorithms on a typical merit function for the design of optical systems. We use two different optical systems to perform the test. They are both eyepieces sharing a number of common specifications. We also compare the performance of the implemented algorithms against the search methods included in OpticStudio (Zemax). We will hereafter abbreviate 'OpticStudio' by 'OS'. We used our own independent raytracing engine to compute the results of the implemented search algorithms.

5.1. Simulation setup

We describe the merit function we used as a measure of performance to optimize in the two optical case studies.

5.1.1. Merit function definition

The MF we defined takes into account two typical optical quality criteria:

- Spot size: Each bundle of rays coming from a given field at the entrance of the optical system must form the smallest possible spot on the image plane. This is an image quality criterion, if it is low then the image will be sharp, otherwise it will be blurry.
- Image position: Each bundle of rays coming from a given field at the entrance of the optical system must focus at a given position on the image plane. This is a control over the focal length as well as the distortion in the optical system.

  Our definition of the MF is then:

$$MF = \frac{\sum_{f=1}^{Nfields} Spot_f + \alpha Pos_f}{Nfields} \tag{9}$$

- *Nfields* : The number of field points taken into account for the computation. We typically sample the object field space with up to a dozen points.

Optics EXPRESS

- $Spot_f$: The spot size criterion for field f , see Eq. (10).
- Pos_f : The image position criterion for field f , see Eq. (11).
- α : A weighting coefficient to balance the two criteria.

$$Spot_f = \frac{\sum_{i=1}^{Nrays} \left(x_i - \bar{x}\right)^2 + \left(y_i - \bar{y}\right)^2}{Nrays} \tag{10}$$

- x_i, y_i : The coordinates of impact of individual rays on the image plane.
- $\bar{x}, \bar{y}$: The coordinates of the centroid of the spot.
- $Nrays$: The number of rays considered in the ray bundle. Typically in the hundreds.

$$Pos_f = \left(\bar{x} - \bar{x}_{trgt}\right)^2 + \left(\bar{y} - \bar{y}_{trgt}\right)^2 \tag{11}$$

- $\bar{x}_{trgt}, \bar{y}_{trgt}$: Target coordinates on the image plane for the centroid of the ray bundle.

5.1.2. Raytracing implementation details

We have developed simple raytracing routines. They perform raytracing on optical systems with surfaces positioned arbitrarily in 3D space and support spherical as well as XY polynomials surface representations. They are limited to monochromatic raytracing, but support the sampling of multiple field points as well as arbitrary aperture stop sampling. We also limited ourselves to the case of aperture stops located in the object space of optical systems. For the two case studies in the present paper, we used an aperture stop sampling consisting of a uniform Cartesian grid cropped by the aperture shape, which is a disk.

5.2. Characteristics of the case studies

We specify the optical characteristics of our two case studies as they would be in a typical optical design preliminary work. The two optical systems share the same set of basic target specifications but they differ greatly in shape and complexity.

5.2.1. Specifications for the two optical systems

We give the optical specifications for the two eyepieces we mean to optimize. The first eyepiece, drawn in Fig. 4, has 5 spherical lenses in an axio-symmetric configuration. This is a typical eyepiece for use with amateur telescopes for example. This system has 21 DOF and is comparatively easier to optimize: it is easy to reach realistic systems with good optical quality. The second eyepiece, drawn in Fig. 5, is a freeform prism with 2 optical surfaces described by fifth order XY polynomials (defined exactly as the surface type “Extended Polynomial” in OS). The whole prism is plano-symmetric. This system has 22 DOF. System 2 is inspired from the general shape of the system in [9]. The optical specifications are found in Table 2. The aperture stop diameter is the diameter of the eye pupil. The focal length is constrained by the field of view (FOV) in the eye space and size of the object. We give the design parameters specific to each system in Table 3. The raytracing for both system is done monochromatic and the refraction indices for the various glasses are indicated in Fig. 4 for system 1 and Fig. 5 for system 2. We note that it is preferable to use curvatures instead of radii of spheres for the specification of spherical surfaces. These quantities are inverse of each other. Indeed, radii give rise to unevaluable MF around their zero value, making a “hole” in the search space that hampers the performance of the search algorithms we implemented.

Optics EXPRESS

Table 2. Common specifications for the two optical systems shown in the present paper.

Aperture stop (eye) diameter D_{pup}	5 mm
Focal length	14.2 mm
Total object size (square shape) $D_{obj} \times D_{obj}$	5 x 5 mm
Full FOV (square shape)	20 x 20 deg

Table 3. Specific design parameters for each optical system.

Optical system	5 spherical lenses ; System 1	Freeform prism ; System 2
Nrays (per field)	253	253
Fields considered	+ 0, 2.5, 5, 7.5 and 10 deg	(0;0), (0;10), (0;-10), (10;0), (10;10), (10;-10), (5;0), (5;10), (5;-10), (7.5;5), (7.5;-5) deg
α MF weighting coefficient Eq. (9)	1	0.1
DOF	All radii of curvature and thicknesses. 21 in total.	1 radius of curvature and 10 XY coefficients per surface. 22 in total.
Variable bounds	[-0.1;0.1] mm-1 for curvatures. [3;15] mm for thicknesses.	[-0.1;0.1] mm-1 for curvatures. [-0.5;0.5] mm for XY coefficients. With normalization radius of 10mm.

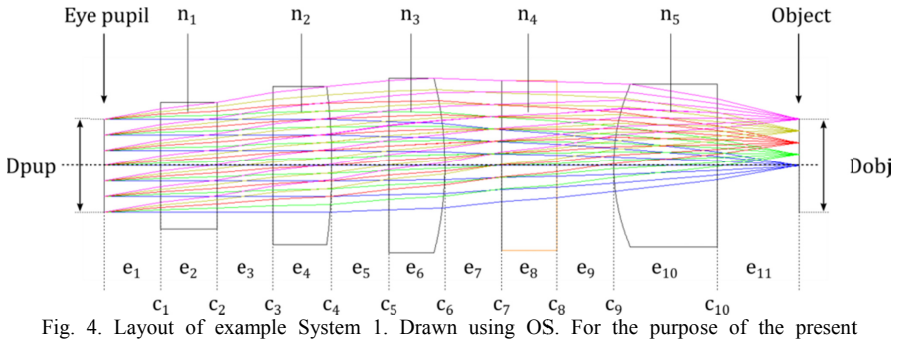


Fig. 4. Layout of example System 1. Drawn using OS. For the purpose of the present simulation, the refractive index are taken invariant with respect to wavelength and n1 = n3 = n5 = 1.52 ; n2 = n4 = 1.64. The optimization variables are the thicknesses e1 through e11 and curvatures c1 through c10, there being 21 variables in total.

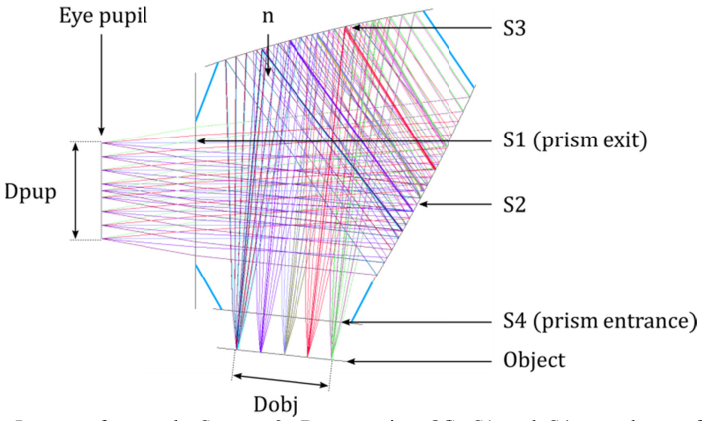


Fig. 5. Layout of example System 2. Drawn using OS. S1 and S4 are plane refractive interfaces. S2 and S3 are reflective surfaces. The refractive index of the prism is n = 1.59. Surfaces S2 and S3 have 11 descriptive parameters which are used for the optimization: curvature of the base sphere and 10 XY polynomial coefficients. We drew blue lines connecting the various surfaces to give the readers an idea of what the prism might look like as a real object.

Optics EXPRESS

5.2.2. Obtaining results on a commercial software for comparison

We recreated the same minimization problems in OS. The surfaces of the optical system are modeled using the native functionalities of the software. The raytracing is also done using the native OS capabilities. The merit function is computed ray by ray using so-called "optimization operands" so as to mirror exactly Eqs. (9)–(11).

To enforce strict bounds on the search space, we resorted to the same type of method as we applied on the CMAES, namely we penalized heavily the MF variables that wander outside the user-defined bounds. We have $MF_{with\_bounds} = MF + bound\_penalty$, the search algorithms in OS ran on MF_{with_bounds}, but we always report MF since it is the true measure of the optical systems' fitness. This may very well have impacted the performance exhibited by the OS search routines. However, we consider the comparison is still fair if the design problem the optical designer sets out to solve is "Find the best optical system according to the MF with system variables strictly within the bounds".

5.2.3. Validation of the raytracing merit function

To assess the exactness of the comparison between our raytracing routines and OS, we compared the values of the MF for 1000 random points within the search space bounds for both of our case studies. The MF is evaluable at these 1000 positions. This test ensures the exactness of our raytracing routines and the concordance between the MF definitions in our routines and in OS.

For each of these 1000 points, we computed the difference $\left|MF_{author} - MF_{zmx}\right|$ between the two results. We report the maximum and mean values of this quantity across our test set in Table 4. The errors are negligible when compared to the lowest MF values found, the lowest MFs for System 1 being in the order of 10^{-5} and for System 2 in the order of 10^{-4} (see Section 5.3).

Table 4. Reported errors in the validation of raytracing implementation and MF on the two example systems.

System	OS vs. own implementation	
	Mean error	Max. error
System 1: Rotationally symmetric eyepiece	5×10^{-10}	1×10^{-8}
System 2: Freeform prism eyepiece	7×10^{-10}	8×10^{-9}

5.3. Results of running the implemented algorithms on the two case studies

We launch, for each case study, 100 independent optimization runs of *at least* 5000 function evaluations for each search algorithm, including those in OS. The OS searches were started with a fixed list of 100 random, evaluable points within the user-provided bounds. We ran two types of search routines provided by OS, which are called: Local optimization Orthogonal Descent (ZLOD) and Hammer OD (ZHOD). The Hammer is a global optimizer. A DLS (Damped Least Squares) mode also exists in OS. However it could not be brought to perform at least 5000 MF evaluations for each optimization run, hence the comparison with other search methods would not have been fair to OS. We report that the performance of the DLS mode appears to be worse than that of the OD mode for both our case studies in test runs that are not reported here. OS provides another search routine called "Global Optimization". We did not test it due to practicality issues. This routine carries out the optimization over many different files, making the results difficult to track.

The results can be found in Fig. 6 and Fig. 7 for the 5 lenses eyepiece (System 1) and in Fig. 8 and Fig. 9 for the freeform prism eyepiece (System 2). The results are formatted in the same manner as those in the BBOB2009 benchmark, except for the y axis of each graph.

Optics EXPRESS

Rather than expressing the ratio of successful number of runs (reaching below Δf_{thresh}), we give the raw number of successful runs. The total number of launched runs is 100 for each search algorithm. The SPX with restart can have a higher number of successful runs, since it has the ability to restart as many times as it wants within its 5000 evaluations budget. We count each convergence of the SPX method with restart as a complete run since it provides each time a viable optical system. The parameterization of the search algorithms for the two case studies is given in Table 5.

Since the global minima of the MF for the two case studies are unknown, we use the raw MF value as criterion for Δf_{thresh}. Namely, we compute $\Delta f = f_{best} - f_{opt}$ with $f_{opt} = 0$, f_{best} being the minimum MF value found so far in a given optimization run.

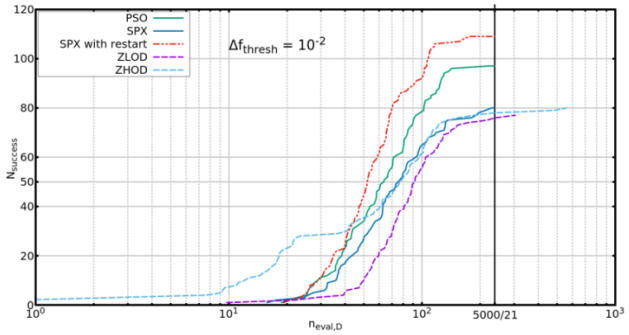


Fig. 6. Result of the implemented algorithms and OS search methods on the MF of System 1. Number of successful runs (out of 100 starting systems) as a function of number of function evaluations per problem dimensionality (21 for System 1). The threshold of the MF value for a run to be considered successful is chosen arbitrarily to be Δfthresh = 10E-2. The thick black vertical line highlights the minimum evaluation budget for every run.

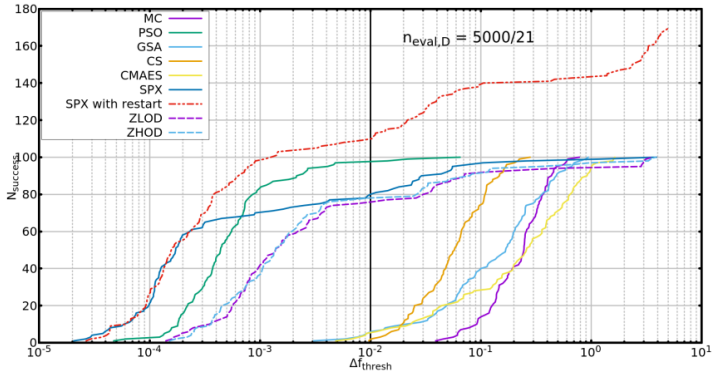


Fig. 7. Result of the implemented algorithms and OS search methods on the MF of System 1. Number of successful runs (out of 100 starting systems) as a function of Δfthresh, the threshold in MF value below which a run is considered to be successful. The function evaluation budget is 5000 for every algorithm. The thick vertical line highlights the Δfthresh value taken in Fig. 6. The number of successful runs for each algorithm at this threshold matches the proportions obtained after 5000 function evaluations in Fig. 6.

Optics EXPRESS

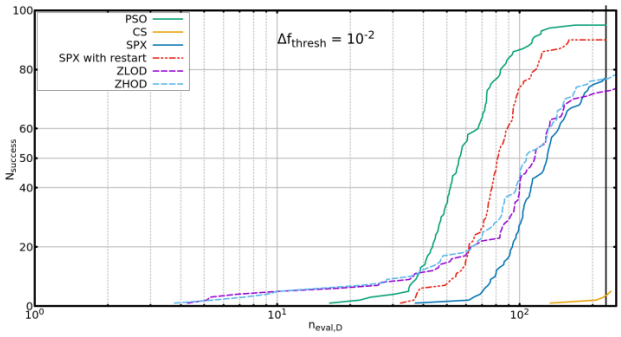


Fig. 8. Result of the implemented algorithms and OS search methods on the MF of System 2. Number of successful runs (out of 100 starting systems) as a function of number of function evaluations per problem dimensionality (22 for System 2). The threshold of the MF value for a run to be considered successful is chosen arbitrarily to be Δfthresh = 10E-2. The thick black vertical line highlights the minimum evaluation budget for every run, which is 5000.

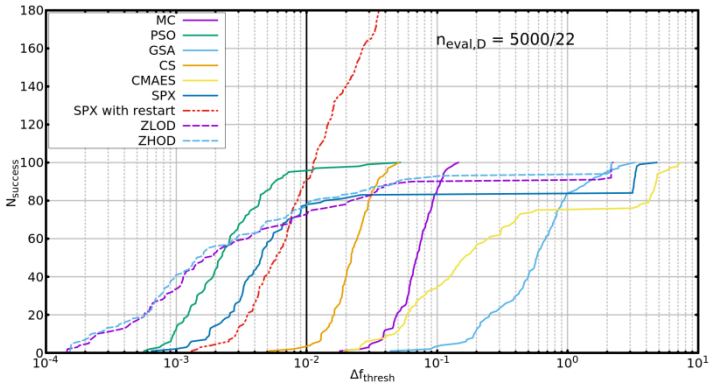


Fig. 9. Result of the implemented algorithms and OS search methods on the MF of System 2. Number of successful runs (out of 100 starting systems) as a function of Δfthresh, the threshold in MF value below which a run is considered to be successful. The function evaluation budget is 5000 for every algorithm. The thick vertical line highlights the Δfthresh value taken in Fig. 8. The number of successful runs for each algorithm at this threshold matches the proportions obtained after 5000 function evaluations in Fig. 8. The total number of runs for the SPX with restart is 292 and not shown on this diagram.

Table 5. Parameterization for each of the algorithms running on the two optical case studies MF.

Algorithm	Parameters used
PSO	$\alpha = 0.7298; \beta = 1.4961; n_{parts} = 20$
GSA	$G_0 = 10; \alpha = 20; n_{parts} = 20$
CS	$n_{parts} = 50; p_a = 0.75$
CMAES	$\sigma = 1$ for system 1 ; $\sigma = 10^{-5}$ for system 2.
SPX	-
SPX with restart	$f_{tol} = 10^{-3}$

Note that whenever an algorithm does not appear in a graph, it means that its performance was too low. Let us analyze the results from the perspective of an optical designer. On Fig. 6, a designer running 100 optimization runs on System 1 would find 50 optical systems with a MF below 10^{-2} in about 50 evaluations per problem dimension (~1050 MF evaluations in total) for the SPX with restart versus about 90 evaluations per dimension (~1890 in total) for the ZLOD. This means that, for the particular problem of obtaining 50 optical systems of a

Optics EXPRESS

given quality out of a 100 tries, the SPX with restart can be seen as a little less than twice as fast as the ZLOD. Likewise on Fig. 8, using the PSO over the ZHOD to obtain 50 optical systems with a MF below 10^{-2} would be about 2 times faster (56 evaluations per dimension for PSO versus 106.5 evaluations per dimension for ZHOD).

Note that, for every example presented here, the computational cost of evaluating the MF was always hugely predominant over the computational cost of managing the searches. This means that the time it takes a designer to run the optimizations is directly proportional to the number of MF evaluations needed.

Now let us look at the ability of the algorithms to reach low global minima, *ie* find better optical systems for the optical designer. If we were to perform a hundred runs of 5000 MF evaluations for each algorithm and look at the quality of the 5 best systems we obtained out of 100, for System 1 on Fig. 7 we would get 5 systems with a MF better than 4.10^{-5} using SPX and 5 systems better than 2.10^{-4} with ZLOD, the 5 best systems of the SPX runs are 5 times better in terms of MF than the 5 best systems of the ZLOD runs. On Fig. 9, for System 2, we can compare the performance of the 5 best systems for PSO and ZHOD. The 5 best systems for PSO are below 8×10^{-4} in MF value, while they are below 1.6×10^{-4} for ZHOD, the ZHOD shows a MF value 5 times better than the PSO.

We note that the SPX that includes a restart strategy is mostly preferable to the SPX without restart strategy. This is not surprising given that a simplex cannot get out of a local minimum. Once the simplex is trapped in a local minimum, it is useless to let it run further. In consideration of this result, the implementation of restart strategies for the other algorithms, detecting similar "stalled" system states, seems very desirable.

To synthesize the present results, for System 1, it is preferable to use the SPX with restart, the SPX or the PSO over the rest of the algorithms. For System 2, the ZHOD and ZLOD are almost always preferable, and the PSO could be used as a complementary method to obtain a greater result variety.

The optical performance of the best found systems in each case study is satisfactory. The best results of System 1 have RMS spot radii of about 5-10 µm and maximum distortion of about 3.5%. For System 2, these values become 10-15 µm and 2.5%. System 2 could for example be used in virtual or augmented reality applications, as it is very small and has a shape compatible with injection molding. The comparison of both case studies highlights the advantages of freeform optics for miniaturization.

The performance of search algorithms depends heavily on the type of problem being optimized. We see this when comparing the relative performance of the search algorithms on the two present case studies. For example, SPX performs very well comparatively to the other methods on System 1, but its performance is underwhelming on System 2. This phenomenon is known as the No Free Lunch theorem of optimization [43]. It states, in layman's terms, that no search method can outperform another search method on all MFs. In fact, if we compare search method A and search method B, there are always as many MFs for which A is better than B as there are for which B is better than A. It would be helpful to the optical design community to study the performance of search methods depending on the types of optical systems and/or on MF definitions.

6. A basic restart scheme for practical use

Using pure search methods facilitates comparisons for the purpose of publishing the present results. However, for practical purposes, there is a great performance gain in using even simple restart schemes, as suggested in [37]. It allows a more user-friendly interface with the search algorithms, as the optical designer can specify parameters like the desired local algorithm (out of the ones in Section 3), the number of best systems to save and display at the end of the optimization process, the evaluation budget of the runs (which can be thought of as

Optics EXPRESS

the time spent in each run), the number of runs etc. An advanced prototyping scheme, closer to the state of the art in general engineering, can be found in [44].

The flowchart for our simple strategy is found in Fig. 10. The general idea is that we first run a "local optimizer", taken from the list of pure search methods, until a stop criterion is met or an evaluation budget is spent. Then we take the result outputted by the local optimizer and run it through a simplex optimizer to reach the bottom of the particular valley we have landed in. We compensate in this way the tendency of some optimizers to hover about the final result. Then we compare the final result with a buffer of past results and save it only if the MF value is good enough and if the system is sufficiently different from the results already stored in the buffer (according to a distance metric between optical systems). The so-called "Global method inputs" are:

- **Merit function**: The function for which we want to have a list of minima.
- **MF variables bounds**: The limits of the search space. The optimizers strictly enforce the limits of the search space.
- **Global budget**: Global merit function evaluation budget for the whole search. We stop the global search once this budget is exhausted.
- **Local budget**: Local merit function evaluation budget for one local optimization round. The local optimizer must stop when its budget is exhausted or when another optimizer-specific stop criterion is reached.
- **Size of results buffer**: This is how many solutions we wish to keep track of and present to the user at the end of the global search. Setting this to 1 makes the global search output only the best global minimum it found.
- **Distance metric threshold**: Threshold below which two points in the solution space are considered the same according to some distance metric and thus not stored twice in the solution buffer. The simplest such metric is the Euclidian norm. But it is not suitable for ill-conditioned problems, and is unsuitable for large dimensionality. A more advanced distance metric between optical systems could be devised by the optical designer on a case by case basis.
- **Other local optimizer specific inputs:** Stop criteria, population size, algorithm parameters etc.

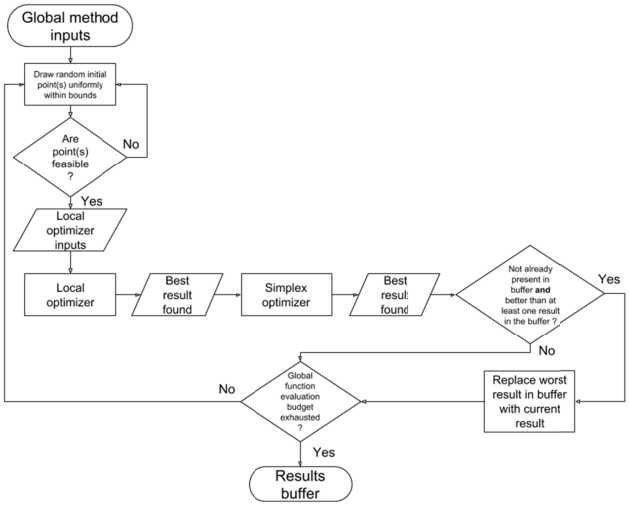


Fig. 10. General flowchart for a global restart scheme strategy.

The output of the method is a list of the best optical systems found that are at the same time sufficiently different from one another according to a provided distance metric. An example of such a list of results running on the MF of System 1 is provided in Fig. 11. We think such a list of results is very useful to the optical designer, as it allows exploration of the

Optics EXPRESS

search space in preliminary design phases. This is a feature that also exists in commercial software. It is called “Global Optimization” in OS.

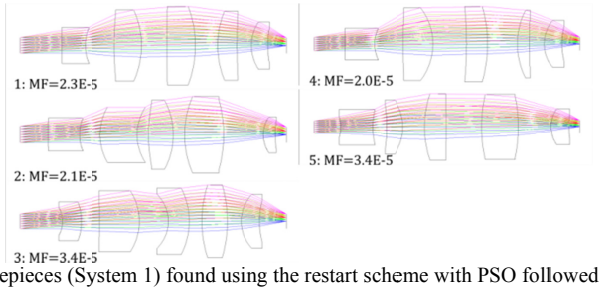


Fig. 11. 5 eyepieces (System 1) found using the restart scheme with PSO followed by the SPX. We use a distance metric of 1E-2 in Euclidian norm to remove duplicate results.

7. Conclusion

We demonstrated useful performance improvements for the search algorithms we implemented against commercial software on optimizing two optical case studies. These performance gains exist despite the poor performance of these search algorithms on a standard benchmark suite when compared to the state of the art. This seems to indicate that further improvements could be made in search algorithms for optical design by applying state of the art search methods. In the future, we think massive improvements could be made by applying state of the art machine learning methods and artificial intelligence to the problem of designing optical systems, particularly in the area of structural changes to optical systems, *eg* adding or removing elements from a system.

We have applied a rigid comparison method between search algorithms (the methodology of the BBOB2009) for given optical systems with given MF definitions. This framework can be applied in future works in order to perform quantitative assessments of the performance of other search algorithms in comparison with the present work.

We have also shown on these two case studies that one should not expect a given search algorithm to outperform all the others for every problem and that it would be beneficial to select the correct algorithm for each problem. We gave an example of a simple restart scheme that can be used in practice by optical designers for global optimization purposes. In combination with internal raytracing software tools that we have developed, we have effectively created a simple yet functional optical design software adapted to higher dimensionality and difficult optimization problems, which include freeform optical design.

8. Technical software details

Software implementations were done using Steel Bank Common Lisp (SBCL) [45] version 1.4.4, which is a Common Lisp compiler. We used OpticStudio Professional 18.7 for System 1 and Premium 18.9 for System 2, belonging to Sophia Engineering. We ask our readers to avoid treating the present results as a rigorous benchmark of the commercial software.

Funding

Sophia Engineering; Thales Alenia Space; Association Nationale de la Recherche et de la Technologie (ANRT).

Acknowledgements

We wish to thank H. Benard and F. Keller at Thales Alenia Space (Cannes, France) for their part in the PhD work. We also thank G. Cassar and M. Guilhem at Sophia Engineering for proofreading this article. We thank especially M. Guilhem for the many hours of discussion about raytracing: We also thank the reviewers of this paper for their insightful remarks.

Optics EXPRESS

References

1. E. Glatzel and R. Wilson, "Adaptive automatic correction in optical design," Appl. Opt. **7**(2), 265–276 (1968).
2. M. J. Kidger, "Use of the Levenberg-Marquardt (damped least-squares) optimization method in lens design," Opt. Eng. **32**, 1731–1740 (1993).
3. D. Shafer, "Global optimization in optical design," Comput. Phys. **8**, 188–195 (1994).
4. K. Höschel and L. Vasudevan, "Genetic algorithms for lens design: a review," J. Opt. **48**(1), 134–144 (2019).
5. S. Banerjee and L. Hazra, "Experiments with a genetic algorithm for structural design of cemented doublets with prespecified aberration targets," Appl. Opt. **40**(34), 6265–6273 (2001).
6. L. Hazra and B. Saswatee, "Genetic algorithm in the structural design of Cooke triplet lenses," Proc. SPIE **3737**, 172–180 (1999).
7. X. Cheng, W. Yongtian, H. Qun, and I. Masaki, "Global and local optimization for optical systems," Optik (Stuttg.) **117**, 111–117 (2006).
8. J. P. Rolland, K. Fuerschbach, G. E. Davis, and K. P. Thompson, "Pamplemousse: The optical design, fabrication, and assembly of a three-mirror freeform imaging telescope," Proc. SPIE **9293**, 92930L (2014).
9. Z. Shen, J. Yu, Z. Song, L. Chen, Q. Yuan, Z. Gao, S. Pei, B. Liu, and J. Ye, "Customized design and efficient fabrication of two freeform aluminum mirrors by single point diamond turning technique," Appl. Opt. **58**(9), 2269–2276 (2019).
10. Q. Meng, H. Wang, W. Liang, Z. Yan, and B. Wang, "Design of off-axis three-mirror systems with ultrawide field of view based on an expansion process of surface freeform and field of view," Appl. Opt. **58**(3), 609–615 (2019).
11. Z. Li, X. Liu, F. Fang, X. Zhang, Z. Zeng, L. Zhu, and N. Yan, "Integrated manufacture of a freeform off-axis multi-reflective imaging system without optical alignment," Opt. Express **26**(6), 7625–7637 (2018).
12. J. Zhu, H. Wei, Z. Xiaodong, and J. Guofan, "Design of a low F-number freeform off-axis three-mirror system with rectangular field-of-view," J. Opt. **17**(1), 015605 (2015).
13. A. Yabe, "Representation of freeform surfaces suitable for optimization," Appl. Opt. **51**(15), 3054–3058 (2012).
14. A. Broemel, H. Gross, D. Ochse, U. Lippmann, C. Ma, Y. Zhong, and M. Oleszko, "Performance comparison of polynomial representations for optimizing optical freeform systems," Proc. SPIE **9626**, 96260W (2015).
15. A. Broemel, C. Liu, Y. Zhong, Y. Zhang, and H. Gross, "Freeform surface descriptions. Part II: application benchmark," Adv. Opt. Technol. **6**(15), 337–347 (2017).
16. M. I. Nikolic, B. Pablo, N. Bharathwaj, A. G. Dejan, L. Jayao and M. Juan Carlos, "Optical design through optimization for rectangular apertures using freeform orthogonal polynomials: a case study," Opt. Eng. **55**(7), 071204 (2016).
17. C. Menke, "Application of particle swarm optimization to the automatic design of optical systems," Proc. SPIE **10690**, 106901A (2018).
18. D. Vasiljevic, *Classical and Evolutionary Algorithms in the Optimization of Optical Systems* (Springer Science & Business Media, 2012).
19. A. Bauer, E. M. Schiesser, and J. P. Rolland, "Starting geometry creation and design method for freeform optics," Nat. Commun. **9**(1), 1756 (2018).
20. J. C. Papa, J. M. Howard, and J. P. Rolland, "Starting point designs for freeform four-mirror systems," Opt. Eng. **57**(10), 101705 (2018).
21. T. Yang, G. F. Jin, and J. Zhu, "Automated design of freeform imaging systems," Light Sci. Appl. **6**(10), e17081 (2017).
22. L. I. Jun, W. Huang, and F. Hongjie, "A novel method for finding the initial structure parameters of optical systems via a genetic algorithm," Opt. Commun. **361**, 28–35 (2016).
23. R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," *Proceedings of the Sixth International Symposium on Micro Machine and Human Science* (1995), pp. 39–43.
24. C. Leboucher, S. Hyo-Sang, C. Rachid, L. M. Stéphane, S. Patrick, F. Mathias, T. Antonios, and K. Alexandre, "An Enhanced Particle Swarm Optimization Method Integrated With Evolutionary Game Theory," IEEE Trans. Games **10**, 221–230 (2018).
25. T. Zeugmann, P., Poupart and J. Kennedy, "Particle swarm optimization," in *Encyclopedia of Machine Learning*, C. Sammut and G. I. Webb, ed. (Springer Science & Business Media, 2011).
26. E. Rashedi, N.-P. Hossein, and S. Saeid, "GSA: a gravitational search algorithm," Inf. Sci. **179**(13), 2232–2248 (2009).
27. S. Özyön, Y. Celal, and T. Hasan, "Incremental gravitational search algorithm for high-dimensional benchmark functions," Neural Comput. Appl. **29**, 1–25 (2018).
28. X.-S. Yang and D. Suash, "Engineering optimisation by cuckoo search," Math. Model. Num. Opt. **1**(4), 330–343 (2010).
29. A. H. Gandomi, X.-S. Yan, and A. H. Alavi, "Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems," Eng. Comput. **29**, 17–35 (2013).
30. X.-S. Yang and D. Suash, "Cuckoo search: recent advances and applications," Neural Comput. Appl. **24**, 169–174 (2014).
31. G.-G. Wang, D. Suash, A. H. Gandomi, Z. Zhaojun, and A. H. Alavi, "Chaotic cuckoo search," Soft Comput. **20**, 3349–3362 (2016).

Optics EXPRESS

32. N. Hansen and A. Ostermeier, "Completely derandomized self-adaptation in evolution strategies," Evol. Comput. **9**(2), 159–195 (2001).
33. "The CMA evolution strategy: A tutorial," 2016, https://arxiv.org/pdf/1604.00772.pdf.
34. N. Hansen, "Benchmarking a BI-population CMA-ES on the BBOB-2009 function testbed," *Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference: Late Breaking Papers* (2009), pp. 2389–2396.
35. A. Auger and H. Nikolaus, "A restart CMA evolution strategy with increasing population size," *2005 IEEE Congress on Evolutionary Computation* (IEEE, 2005), pp. 1769–1776.
36. J. A. Nelder and R. Mead, "A simplex method for function minimization," Comput. J. **7**, 308–313 (1965).
37. Luersen, A. Marco, L. R. Rodolphe, and G. Frédéric, "A constrained, globalized, and bounded Nelder–Mead method for engineering optimization," Struct. Multidis. Optim. **27**, 43–54 (2004).
38. W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes 3rd edition: The Art of Scientific Computing* (Cambridge University, 2007).
39. N. Hansen, F. Steffen, R. Raymond, and A. Auger, "Real-parameter black-box optimization benchmarking 2010: Experimental setup," Research report RR-7215, INRIA, 2010.
40. COCO, (COmparing Continuous Optimisers) homepage, (INRIA, 2019), http://coco.gforge.inria.fr/.
41. N. Hansen, F. Steffen, R. Raymond, and A. Auger, "Real-parameter black-box optimization benchmarking 2009: Noiseless functions definitions," Research report RR-6829, INRIA, 2009.
42. N. Hansen, A. Auger, R. Ros, S. Finck, and P. Pošík, "Comparing results of 31 algorithms from the black-box optimization benchmarking BBOB-2009," *Proceedings of the 12th Annual Conference Companion on Genetic and Evolutionary Computation* (2010). pp. 1689–1696.
43. D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," IEEE Trans. Evol. Comput. **1**, 67–82 (1997).
44. A. Hagg, A. Asteroth, and T. Bäck, "Prototype discovery using quality-diversity," *International Conference on Parallel Problem Solving from Nature* (2018), pp. 500–511.
45. Steel Bank Common Lisp (SBCL) homepage, (2019), *http://www.sbcl.org/*.

Erratum
Vol. 27, No. 20 | 30 Sep 2019 | OPTICS EXPRESS 28383
Optics EXPRESS

Comparing optimization algorithms for conventional and freeform optical design: erratum

THOMAS HOULLIER,[1,2,*] AND THIERRY LÉPINE[1,3]

[1]*Univ-Lyon, Laboratoire Hubert Curien, UMR CNRS 5516, 18 rue Benoît Lauras, 42000 Saint-Etienne, France*
[2]*Sophia Engineering, 5 Rue Soutrane, 06560 Sophia Antipolis, France*
[3]*Institut d'Optique Graduate School, 18 rue Benoît Lauras, 42000 Saint-Etienne, France*
**thomas.houllier@sophiaengineering.com*

Abstract: We want to acknowledge prior research with a very similar focus to our own which we failed to discover before the publication of our paper.

A research paper [1] with very similar goals to our own research has been brought to our attention. It was published some months before we sent the manuscript for the present paper. Much like our own paper, it compares optimization algorithms on optical case studies. We think we needed to provide proper acknowledgement of prior research in the present erratum.

The comparison methodology, optical systems presented and optimization algorithms used differ between the two papers however.

References

1. F.E. Sahin, "Open-source optimization algorithms for optical design," Optik **178**, 1016–1022 (2019).

#376681
Journal © 2019
https://doi.org/10.1364/OE.27.028383
Received 4 Sep 2019; published 19 Sep 2019

7.4 Going further

A lot of practices that are usual in the optimization literature or in other engineering disciplines are left underexploited in optical design. These include hybrid optimization and decision-making (Fig. 7.3), surrogate optimization (Fig. 7.4) and modeling using deep learning, and finally multi-objective (or *Pareto*) optimization (Fig. 7.5).

7.4.1 Hybrid Optimization and Decision-Making

We have mainly touched upon real-valued continuous variables in the present chapter, because this is the main type of variables in optical design, particularly reflective telescope design. However, glass selection is an important part of lens design. Glass selection is a discrete optimization problem since the designer is constrained by the glass catalog of glass manufacturers (see the websites of Schott, Ohara, CDGM, *etc*). Glasses, to a sufficient approximation, are discrete points in a 2-space constituted by refraction index and Abbe number. Search problems mixing continuous and discrete parameters are often called *hybrid*. They can be solved with algorithms such as Ant Colony Optimization (ACO) [195] (here applied to optical design [196]). This type of optimization exists in commercial software as "Glass Substitution" (OpticStudio) and "Glass Expert" (CodeV).

Another type of optimization, related to *discrete* or *combinatorial* problems, is that involving *decisions*. An applicable case in optical design is that of adding or removing elements in the design. This decision is typically made by the optical designer based on how far from the performance requirements a design is after many attempts at optimization. A designer will typically add elements if the performance remains poor and remove elements if the performance is excellent (so as to cut cost and unnecessary complexity)[6]. Such problems seem related to *Markov Decision Processes* [197], although we have not found it applied specifically to optical design and have not tried it ourselves.

7.4.2 Surrogate Optimization

There are many ways to assess the performance of an optical system, which trade accuracy for computational cost. The main simulation type we have talked about in the present work is sequential raytracing of "real rays", *ie* applying Snell laws on an aperture stop sampling through a cascade of diopters from the scene to the detector. However, there are other simulation types we can take advantage of. These could integrate well with *surrogate optimization* (an overview of which can be found in [198]). The general idea in surrogate optimization algorithms is to use at least an accurate but costly simulation model and an approximate but cheaply-computed model. The cheap model is evaluated more frequently than

[6] This is at least the simplest way of looking at the problem of the number of elements. Other considerations such as lens groups in zoom system are equally important but would be more difficult to model.

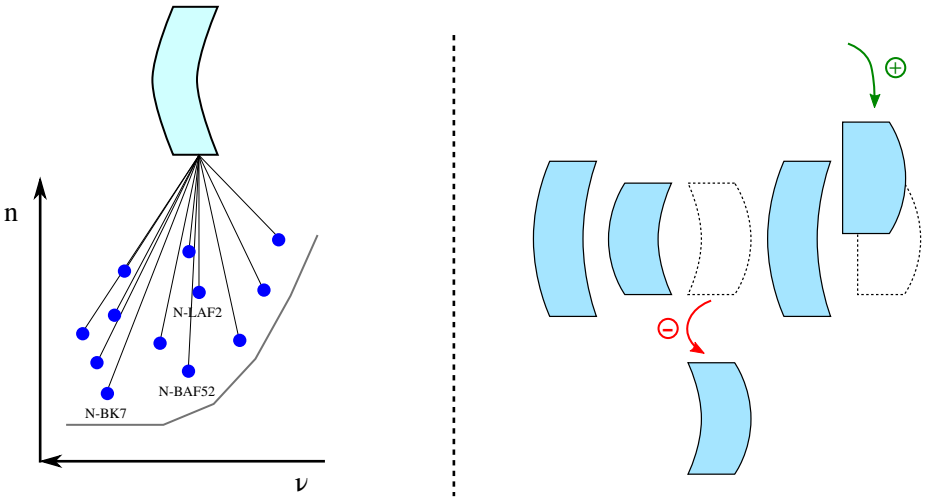


Figure 7.3: Alternative variable types: glass selection is an example of discrete variable and the search process could also include decision-making about adding or subtracting elements.

the costly model and is used to guide the general direction of the search in order to find minima in the costly function landscape quicker.

The different broad types of optical simulation we can use are, in rough order of increasing accuracy and computational cost are (in the style of Fig 1 in [198]):

1. Low order aberration theory for spherical systems (*eg* Seidel).
2. Real raytracing.
3. Freeform aberration theory (*eg* NAT).
4. Ray-based Electro-Magnetic (EM) descriptions (*eg* Fresnel formulae).
5. Straylight analysis based on empirical models (BRDF *etc*).
6. Wavefront propagation models (*eg* Fourier optics).
7. Experiment.

Tolerancing results could be used as a performance metric. Tolerancing can be performed at any of the listed simulation levels (although tolerancing on top of a straylight analysis would have a prohibitive cost). Note that quickly-evaluated shortcuts for as-built tolerances exist [199].

As a sidenote, machine learning models have been demonstrated that successfully take specifications as inputs and output a vector of parameters describing optical systems [200, 201]. These models are trained on existing sets of optical systems and corresponding specifications.

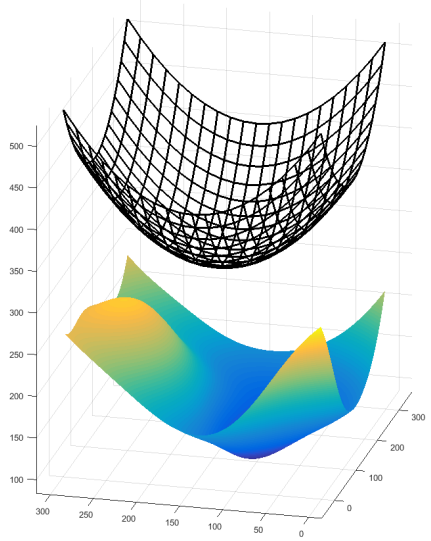


Figure 7.4: Surrogate optimization: illustration. An accurate but computationally costly model is approximated by an alternate rough but cheaply-computed model. Surrogate optimization strategies make use of both.

7.4.3 Pareto Optimization

An overview for *Pareto optimization* (also named *multiobjective optimization*) is provided in [202]. It involves the simultaneous optimization of several objectives. The result is not a single vector of parameters but several, called *Pareto optima* (plural), along the *Pareto front*. The Pareto optima are vectors of parameters that cannot be modified to improve one objective without worsening at least one other.

Performing Pareto optimization is advantageous in optical design (as well as in many other engineering disciplines) because of the Pareto optimal systems it provides. Obtaining a set of Pareto optima is better than obtaining a single system that combines all the objectives into one (such as with a typical MF) with an arbitrary weighting. The reason is that engineers can see the available trade-offs and orient the design accordingly.

We think that search algorithms are underexploited in optical design. By necessity and lack of research effort due to the smallness of the community, optical designers rely on existing tools. We hope to have shown how much larger both the use and study of search algorithms really is.

It seems to us that optical design is a field with many low-hanging fruits for applied mathematicians to pick. The functions generated in optical design are very hard to optimize while being quite fast to evaluate. These properties make

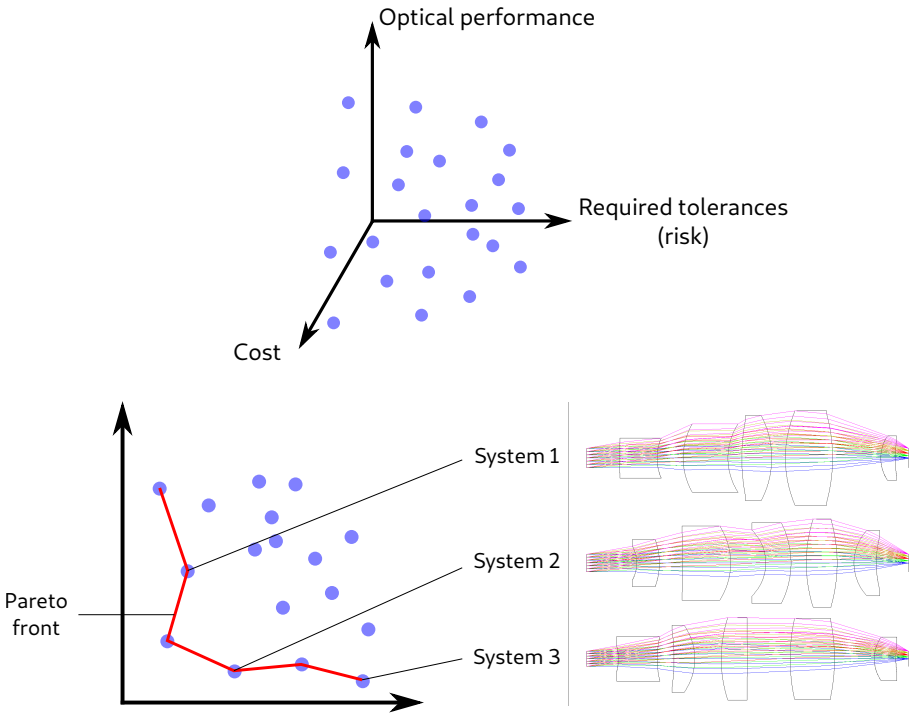


Figure 7.5: Illustration for multiobjective optimization and Pareto optima: what it could look like in optical design.

them, as far as our understanding goes, appropriate challenges, in themselves, for search algorithms. Research could have been hampered by a general lack of communication between optical designers and applied mathematicians or by the difficulty in finding convenient and available software tools to both communities. In any case, we hope this interdisciplinary research effort, to which we have made a small contribution, can grow in the future and cover the increasing need generated by more and more complex freeform systems.

Chapter 8

Freeform metrology overview: shape measurements

Measurands – Measurement geometries – Sampling – Ranges – Duration – Environment – In-situ – Coordinate Measuring Machines – Chromatic Confocal Microscopy – Contact Stylus Profilometry – Curvature sensing – Null interferometry – Shearing interferometry – Shack-Hartmann – Phase-Measuring Deflectometry – Comparison Matrix

In this chapter, we review metrology methods suitable for the measurement of freeform optical surfaces. The difference between *conventional* and *freeform* optics metrology is quantitative: any surface is freeform at a small enough altitude scale. However, freeform optical surfaces exhibit large, intended, departures in altitude and slope from reference flats and spheres. This prompted the development of new methods and instrument architectures. The gold standard that was, and still is, interferometry for planes and spheres is less applicable to freeform surfaces without costly and difficult to align nulls, be they lenses, mirrors or Computer-Generated Holograms (CGH). There are many, available or promising, alternative measurement methods. The scope of the review is narrowed to:

- *Continuous* surfaces in altitude, preferably in slopes too.
- *Shape* measurement (low to mid spatial frequencies), we do not mention roughness measurement.
- Surfaces with a roughness sufficient to obtain a *specular* reflection.

We keep the present review short and general, the idea is to have sufficient information to be able to interact with metrology experts and manufacturers.

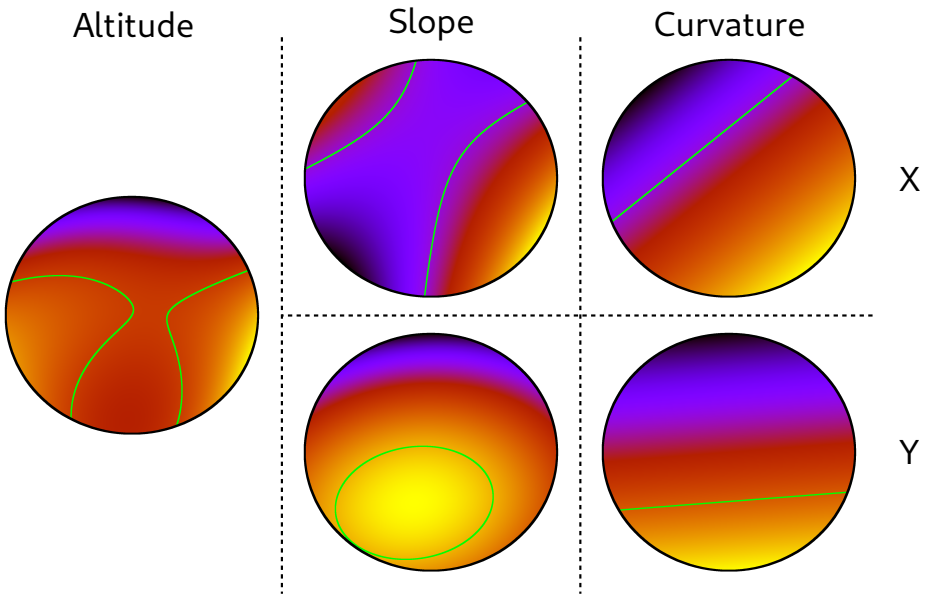


Figure 8.1: Illustration: altitude, first derivatives and curvatures in X and Y directions for a random third order XY freeform surface. The green contour line is the zero in each heatmap.

We refer the reader to other reviews of freeform optical metrology [1, 41, 203, 204].

8.1 Criteria for measurements of surfaces

Let us list some of the comparison criteria (in a non-exhaustive way) for metrology methods. These help deal adequately with trade-offs in the often more difficult measurement of freeform optical surfaces.

8.1.1 Measurand

Measurement methods can be classified based on their measurand:

- Altitude
- Slope
- Curvature

The methods with an *altitude* measurand will provide the surface shape with minimal additional data processing (at least in principle, ignoring calibration issues that can account for a significant part of a method's complexity). Methods with a *slope* or *curvature* measurand require an additional integration step

[205–207] that can magnify the measurement errors. However, *slope* and *curvature* measurands are less sensitive to errors in reference frame calibration [208], curvature in particular being entirely intrinsic to the surface (does not depend on the measurement reference frame). We illustrate the different measurand on Fig. 8.1. Note that we need *at least* two distinct measurement directions for slopes and curvatures.

8.1.2 Measurement geometry

Another high-level discrimination criterion between metrology methods is the measurement geometry. The different architectures (illustrated in Fig. 8.2) are:

- *Point-based*. A probe, whether mechanical, optical or electronic, is mounted on an arm scanning the optical surface point by point. Some of these systems are called Coordinate-Measuring Machine (CMM).
- *Full-field*. The entirety of the surface aperture is measured in a single acquisition step.
- *Stitched aperture*. The metrology systems is only able to measure subapertures of the surface, these subapertures are *stitched* together to form a reconstructed whole aperture result.

Full-field measurement requires, when applicable, only a single calibration of the relative position of the measurement system to the surface. In addition, measuring the position of the surface relative to fiducials is simpler since they can, in some cases (*eg* deflectometry and CMM), be included in the measurement data directly.

Stitching is required when the instrument has a field that is smaller than the sample aperture. This can be the result of a trade-off that favors a finer lateral resolution. Stitching requires a careful calibration and data processing [209].

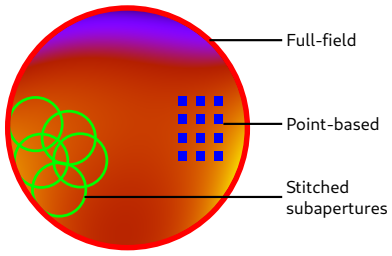


Figure 8.2: Illustration: Measurement geometries in metrology systems.

Point-based machines or indeed stitching methods require moving a probe or an optical head relative to the sample. This can be achieved by rotating the sample on a spindle and/or moving the probe. This makes calibration necessary at each position and requires even more attention to the mechanical structure of the instrument (see the NANOMEFOS [210]).

8.1.3 Sampling accuracy & resolution

Metrology systems can be characterized by their surface sampling accuracy and resolution, both lateral and in altitude.

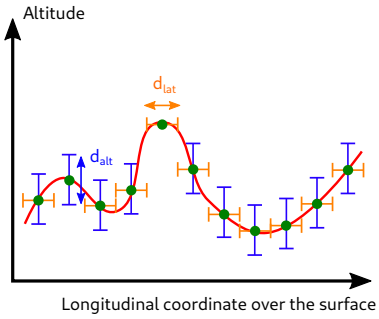


Figure 8.3: Illustration: Sampling a surface shape with a given lateral sampling step d_{lat} and an altitude accuracy d_{alt}.

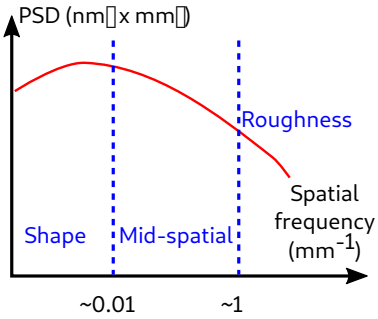


Figure 8.4: Illustration: PSD. The lowest spatial frequencies correspond to shape, then come the mid-spatial and roughness ranges. We are concerned mostly with shape and mid-spatial frequencies in the present chapter.

8.1.3.1 Lateral sampling

We generally use the range of Power Spectral Density (PSD) frequencies to characterize metrology instruments [63, 203], as illustrated in Figs. 8.3 and 8.4.

There is often a trade-off between the measurement field and the lateral resolution (*eg* in interferometry), or alternatively between lateral resolution and measurement duration for most CMMs.

8.1.3.2 Altitude sampling

This criterion has to be compared with the design requirements of the manufactured optical surfaces and the results of tolerancing analyses. The required accuracy is largely driven by the application domain: telescopes in the near-Ultraviolet (UV) and visible wavelengths have stringent surface shape requirements (*eg* 1 nm RMS), while IR systems will require less accurate metrology (*eg* 10 nm RMS).

8.1.4 Measurement ranges

Metrology systems have an allowable range of measurement slopes and altitude.

8.1.4.1 Slope limit

The range of measurable slopes (Fig. 8.5) is often a limiting factor in freeform metrology, especially for full-field measurements (*eg* CGH null-interferometry). The CMM-type machines are less impacted as the measurement head can often be rotated to follow the slope of the sample.

With regards to general surface shape, concave samples are generally easier to measure than convex ones. For example, the access to a focal point of a more

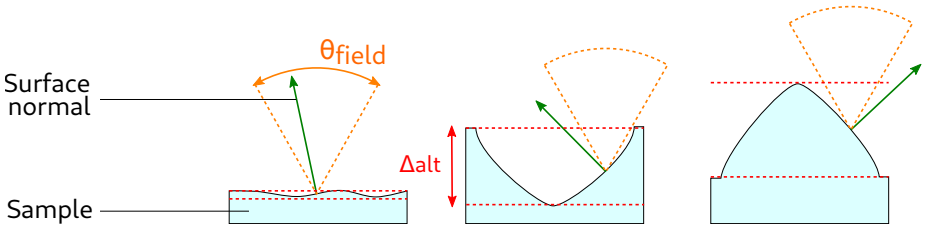


Figure 8.5: Illustration of the measurement ranges in altitude and slope. Metrology methods are usually limited to measuring slopes within some cone θ_{field} and inside some altitude range Δ_{alt}.

or less spherical wavefront allows very straightforward setups (*eg* Fizeau interferometer with spherical reference). Convex samples generate problems for optical measurements since the light is likely to be reflected out of the measurement system (*eg* interferometry, deflectometry).

Similarly, a limit in curvature also exists: *eg* the radius of a stylus probe. This can lead to filtering in the measurement that is hard to detect *a posteriori*[1].

8.1.4.2 Altitude range

The range of measurable altitude, either in an absolute sense or deviating from a null (*eg* interferometry), is a commonly limiting factor in freeform metrology.

8.1.5 Measurement duration

Particularly for CMM machines and stitching setups, there is an obvious trade-off between the quantity of data (which can increase the lateral resolution or the robustness of the data) and the measurement duration. The reasons why we might want to keep the measurement duration short are: cost in man-hours supervising the measurement, long measurements impede the back and forth between manufacturing and metrology, calibration shifts (*eg* thermal) can happen during the time that separates the calibration step from the acquisition itself *etc.*

8.1.6 Environmental perturbations

The susceptibility of a measurement setup to environmental perturbations is a factor in ensuring reliable results. These perturbations are typically:

- Vibrations.
- Ambient lighting.
- Dust and particles.

[1] See also an example of a measurement bias between *average slopes* (lateral shearing interferometer) and discretely sampled slopes (deflectometry) [208].

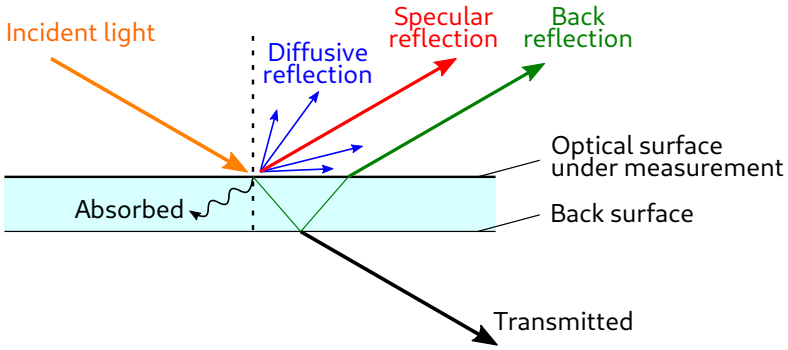


Figure 8.6: Breakdown of incident light on a refractive sample for optical metrology systems.

- Temperature.
- Air convection.

Additionally, for optical-based systems, the level of sample surface scattering (linked to surface roughness) as well as reflectivity and back reflections are other sources of perturbations (Fig. 8.6).

8.1.7 Portability and in-situ measurements

System size is important in particular for in-situ measurements. The conventional workflow for manufacturing optical surfaces is to execute successive machining passes guided by a measurement in between each one. The back and forth between manufacturing machine and metrology instrument is time-consuming and generates repositioning problems. A possible solution is to integrate the metrology instrument in the manufacturing machine. Such integration is however difficult and is the subject of ongoing research [211, 212] and programs[2].

8.2 Some shape measurement methods

We present a non-exhaustive list of metrology methods relevant to freeform optics and explain succinctly their working principle.

8.2.1 Coordinate Measuring Machines

We classify in this section methods which scan a point-based measurement probe above the sample. The probes can be of various types: some are *contact* probes and some are *optical*. In addition to the more traditional methods we present in this section, the reader may read the following papers [213–215] on an optical probe-based instrument using cascade optical coherence tomography, provided

[2]see ESA ITT AO/1-10040/19/NL/AR "METROLOGY FOR OPTICAL FREE-FORM SURFACES"

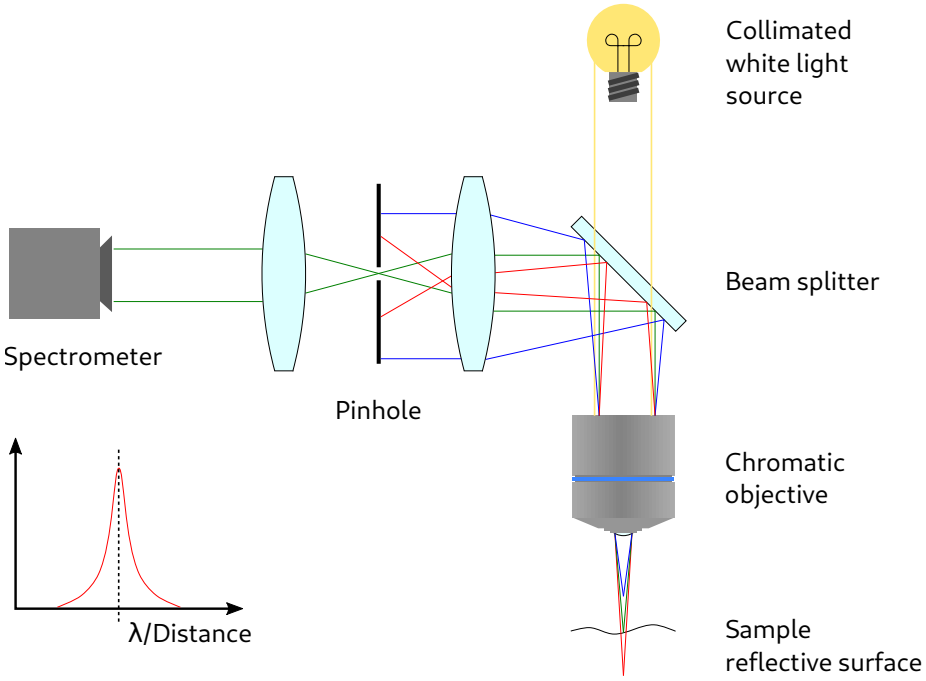


Figure 8.7: Chromatic Confocal Microscopy working principle.

with an assessment on a freeform mirror with shape error residuals in the dozens of nm RMS.

8.2.1.1 Optical probes: Chromatic Confocal Microscopy

Let us take chromatic confocal microscopy as an example of non-contact optical probes, although many others exist (multi-wavelength interferometry in LUPHOScan, laser triangulation *etc*). Chromatic confocal microscopy makes use of an objective lens with deliberately large longitudinal chromatism (Fig. 8.7). By sending white light through this objective lens, light is focused at different altitudes depending on the wavelength. The sample surface reflects all the light back through the objective and into a pinhole system. The pinhole selects the wavelength that was focused around the surface altitude. A spectrometer is then used to further narrow down which wavelength was better focused on the sample surface (peak intensity).

The dynamic range of measurable altitudes is determined by the bandwidth of the white light and the magnitude of the longitudinal chromatism of the objective lens. The resolution in altitude is determined by the resolution of the spectrometer and the magnitude of the longitudinal chromatism. For example, a chromatic probe working with wavelengths ranging from 300 nm to 1300 nm, with a target of 100 nm in altitude resolution would require a spectrometer with

Altitude range	1 mm
Slope limit	± 44°
Accuracy	100 nm
Resolution	50 nm
Spot size	8 µm
Max. acquisition freq.	10 kHz

Table 8.1: STIL EVEREST K1 Optical pen specifications [217].

Travel dimensions (X × Y)	200 mm × 700 mm
Lateral resolution	125 nm (X axis)
Lateral accuracy	> 1 µm (X)
Altitude range	5 mm
Altitude resolution	0.4 − 20 nm
Altitude repeatability	100 nm
Tip radius	2 µm

Table 8.2: Specifications for Taylor Hobson Form Talysurf CNC series profilometer [219].

$\frac{1300-300}{10^6/100} = 0.1$nm (in the naive hypothesis where the dynamic range comes from the probe alone). We list the characteristics of a typical commercial sensor in Tab. 8.1. Note that probes that acquire multiple points on a line also exist. See also a related interferometric setup [216].

8.2.1.2 Contact Stylus Profilometry

Contact stylus profilometry is a widespread point-based surface shape measurement method. A mechanical probe (the technology of which is in itself quite involved and for which many alternatives exist) is typically mounted on an arm above the sample and scanned across the surface, point-by-point or along lines.

Stylus profilometry is very flexible due to the usually quite large altitude and lateral measurement range. The altitude range is furthermore combined with a good altitude resolution. It does not require the surface to be reflective. In addition, the optical surface as well as the surrounding mechanical parts can be measured together. The main drawbacks are: the surface is likely to get scratched by the probe, the method is point-based and the measurement duration is usually long, the acquired surface profile is biased by the fact that the stylus tip is not punctual [218] and soft materials can also bias the measurement. The specifications of a commercial system are given in Tab. 8.2.

8.2.1.3 Curvature Sensing

As noted by Rose [220] in their review, an instrument was developed for point-by-point curvature sensing at NIST [221]: the GEMM (GEometry Measuring Machine) [222, 223]. Its working principle is as follows: an optical head is scanned along the sample. Inside the optical head, an interferometer with a small aper-

ture measures an altitude map locally. From this map are extracted curvature values for the current lateral position of the head above the sample. In this fashion, a point-by-point grid of curvatures is acquired. These curvatures are then integrated twice to obtain the altitude [224]. The advantage of measuring curvatures is that this measurand is intrinsic to the surface shape, thus calibration requirements are looser. The curvature measurements, being computed from an interferometric map with many pixels, also exhibit the very low noise required by the double integration operation.

The reported repeatability of GEMM is 3 nm RMS, and a cross-validation on a test surface against another metrology method shows a match within 6 nm RMS [223]. Another curvature sensing setup with the same working principle can be found in [225]. See [226] for a Shack-Hartmann based curvature sensor which outputs local principal curvatures and principal directions.

8.2.2 Null Interferometry

Interferometry is traditionally the main metrology method for precision optics manufacturing, typically Fizeau setups with reference flats or spherical references. Freeform surface testing however, when the test wavefront differs significantly from a sphere (the spatial frequency of the fringes becomes too high to measure), requires freeform nulls in order to generate a freeform reference wavefront.

The *null* can be:

- A CGH [11, 227–229] including some slightly adaptive setups [230].
- A lens or optical system [231].
- A Spatial Light Modulator (SLM) or Deformable Mirror (DM) [232–235].

Null interferometry has the advantage of being well understood by a large number of companies and labs. The achievable resolution and accuracy are in the nanometer range. Some of the limitations are: *a priori* knowledge of the sample shape is required, a different null must be manufactured for each surface under test (in the case of non-adaptable nulls), there is usually some slope limit to the phase of the null (*eg* attainable slope in CGH phase is limited by the manufacturing process), nulls are sensitive to alignment errors.

8.2.3 Shearing Interferometry

Shearing interferometry is a full-field slope measurement method. It uses two almost coincident wavefronts separated on the sample by a lateral distance called *shear distance*. In contrast with classical interferometry, there is no reference wavefront, the two wavefronts hit the surface sample but at slightly different positions. The difference in fringe phase between interferograms with two different, known, shear distances is computed. This working principle, which relies on

measuring the variation of phase and not the phase directly, is why this method is called *differential interferometry*.

For a lateral shearing interferometer with displacement in two orthogonal directions x and y, one can establish [220] Eq. 8.1:

$$\begin{cases} \alpha_x = \Delta\Phi_x \cdot \frac{\lambda}{4\pi \cdot \delta_x} \\ \alpha_y = \Delta\Phi_y \cdot \frac{\lambda}{4\pi \cdot \delta_y} \end{cases} \tag{8.1}$$

With:

- λ: Measurement wavelength.
- α: *Average* surface slope across the shear distance.
- δ: Lateral shear distance on the sample.
- $\Delta\Phi$: Measured phase variation between zero shearing and applied shear distance.

Taking the Michelson configuration in [220] as an example, a shear angle is introduced using piezo-electric actuators on one of the mirror (phase-shifting is also performed using these actuators). A possible measurement method is to measure the phase difference $\Delta\Phi$ between a reference phase at zero shearing, and a phase measured via *temporal phase-shifting* with a given, known, *shear distance* δ. The shear distance can be calibrated using a grid in lieu of the sample.

Possible variations include:

- Scan with the wavelength instead of the shear distance [236]. This eliminates the need for a potentially inaccurate or vibration-generating mechanical actuators.
- Use polarisation to add a calibration arm to the setup [236].
- Use several shear values and directions (*eg* multilateral shearing interferometry) [237].

Given that the two interfering beams share almost the same path in space, shearing interferometry setups generally have a low sensitivity to vibrations. Moreover, the slope measurement range can be adjusted via the shear distance, making the instrument flexible.

8.2.4 Shack-Hartmann

Shack-Hartmann sensors Fig. 8.8 measure the slopes of an incoming wavefront using an array of microlenses [238, 239]. It is a full-field slope-measuring method. Through each microlens, the local wavefront is focused on a sensor, creating a spot, the position of which is related to the local slope. Calibration to account for the actual aberrations in the microlens array is necessary.

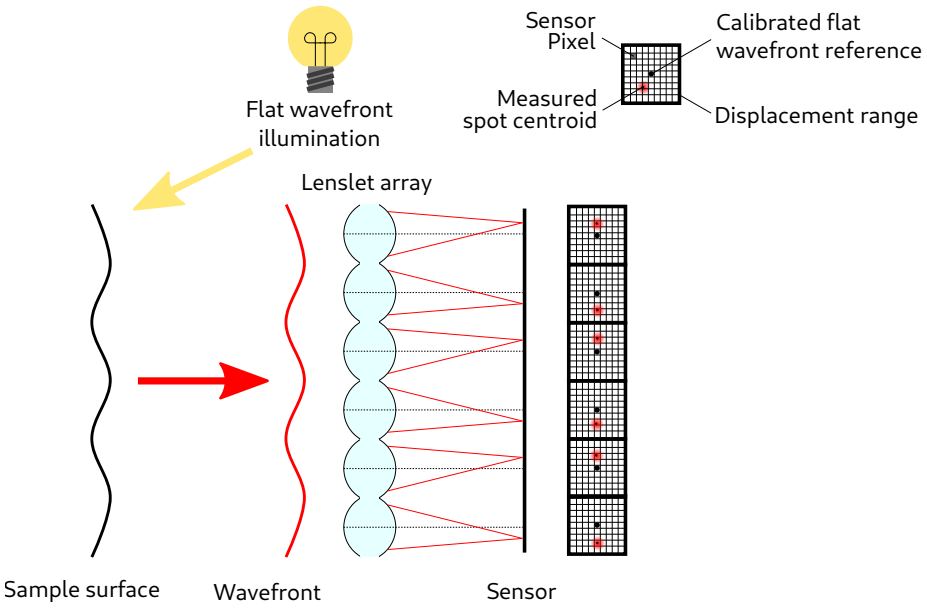


Figure 8.8: Shack-Hartmann working principle.

The lateral resolution is limited by the number of microlenses in the array. There is only one data point per microlens in a single acquisition. The tilt dynamic range is also quite narrow since spots must land on the sensor in the right "channel" (the area of the sensor dedicated to the microlens the light passed through), although it is possible to track spots across channels with unwrapping algorithms [240, 241] or have a scanning setup [242] to increase the measurement range. We give the specifications of a commercial system in Tab. 8.3. One advantage for Shack-Hartmann systems is that they can be packaged in quite compact camera-format systems with factory calibration.

Aperture dimension	$3.6\,\text{mm} \times 4.6\,\text{mm}$
# of microlenses	32×40
Lateral sampling resolution	$110\,\mu\text{m}$
Tilt dynamic range	$\pm\ 3°$
Tilt sensitivity	$5\,\mu\text{rad}$ RMS
Wavefront accuracy	$\approx \lambda/100$
Acquisition frequency	$\approx 100\,\text{Hz}$
Working wavelength	$400 - 1100\,\text{nm}$

Table 8.3: Imagine Optic HASO4 First specifications [243]. λ is a single wavelength chosen within the working range and for which the system is calibrated.

8.2.5 Phase-Measuring Deflectometry

Phase-Measuring Deflectometry (PMD) is a full-field slope measurement method [244–250]. A camera observes fringes from a display screen reflected on a specular surface. With careful calibration it is possible to measure the phase of a known fringe pattern as reflected by the sample, and translate it into slopes. Once a slope map is obtained, integration can be performed and an altitude map obtained. An additional piece of prior information is required though: the position of one point on the surface in the reference frame of the setup. This is to solve the *slope/height ambiguity*: the solution surfaces that reflect rays from the screen towards the camera are a continuous set of sample positions and shapes. The ambiguity is broken by constraining the solution surface to pass through a known point in space (more in Chapter 9).

The accuracy of PMD is limited by the knowledge of the geometry of the setup, *ie* by the quality of the required calibrations. The measurement range, both in sample size and sample slopes is potentially very large, it is only limited by the size of the display: a measurement can be carried out as long as the FOV of the camera through the sample lands within the limits of the display screen (Fig. 8.9). The lateral sampling is limited by the camera sensor and the achievable size of the sample in the camera FOV (with the additional hypothesis that the display pixels are unresolved by the camera).

There is still ongoing research in taking into account more and more factors in the calibration of PMD in order to reach better accuracy, for instance in the display screen model: both shape and refraction from the protective layer on top of the pixels can be taken into account [251, 252].

An example of an application using a SCOTS (Software Configurable Optical Test System) instrument on the measurement of a Giant Magellan Telescope (GMT) primary mirror segment (8.4 m diameter) with 25 nm RMS agreement with interferometry is found in [253], other results in [254]. Deflectometry, due to the simplicity of the required equipment, is often considered for in-situ measurements [211]. It is also very suited to appearance defects measurements and semi-quantitative measurement of surfaces [255].

Notable variants include:

- Transmission setups [256].
- Application to discontinuous surfaces [257].
- Setups with more cameras and/or screens [258, 259].

A related method is that of *fringe projection*, for the measurement of diffuse surfaces [260–262].

8.3 Comparison Matrix

We can summarize the above information with a high-level comparison matrix (Tab. 8.4).

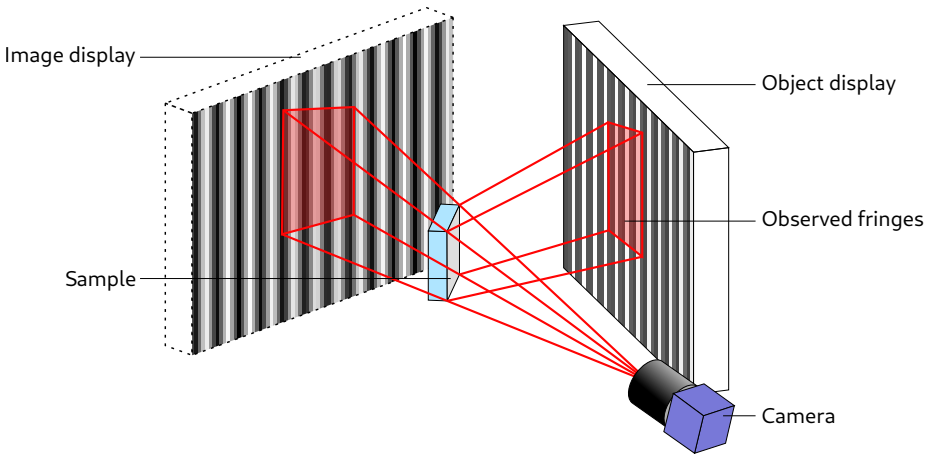


Figure 8.9: Illustration for the measurement range of deflectometry. The camera looks at the display screen through the sample (here nearly flat). The field of view of the camera through the sample is drawn as the red solid. The intersection of the solid with the screen is the surface of the screen observed by the camera. As long as this surface is where fringes can be displayed, a measurement can be performed.

Method	**Measurand** **Geometry**	**Advantages** **Limiting factors**
Chromatic Confocal Microscopy	Altitude Point(s)	Slope range Z range/resolution trade-off
Contact Stylus Profilometry	Altitude Point	Z range/resolution trade-off Diffuse Scratches, Duration Measurement bias
Curvature Sensing	Curvature Point	Intrinsic, calibration Noise
Null Interferometry	Altitude Full-field	Accuracy Cost, Alignment, Reference
Shearing Interferometry	Slope Full-field	Dynamic range Calibration
Shack-Hartmann	Slope Full-field	Compact Lateral sampling
Phase Measuring Deflectometry	Slope Full-field	Slope range, Cheap Calibration

Table 8.4: Comparison matrix of metrology methods.

This past chapter has shown there are many measurement methods for freeform optical surfaces. As with many things in technology, the best method will very much depend on the use case. Need to control the shape of thousands of identical surfaces on an automated assembly line? Null interferometry might be indicated. Need to measure the shape of objects with somewhat diverse surface shapes and contours, as quickly as possible? More flexible methods such as deflectometry or shearing interferometry will be prefered. Need to assess the position of an optical surface with respect to surrounding non-reflective mechanical parts? Contact Stylus Profilometry is the best tool for this. In the following chapter, we will perform measurements of freeform optics using Phase Measuring Deflectometry.

Chapter 9

Metrology experiments

Designing a freeform benchmark test – Deflectometry bench and setup – Different types of results – Results on the freeform piece: cosmetic, curvatures, absolute shape, mid-spatial frequency content – Other samples

To demonstrate the ability of PMD to measure challenging optical surfaces, including freeform, we performed measurements on a freeform test piece and other various samples. The measurements on the freeform test piece were both qualitative (appearance defect detection) and quantitative (shape measurements).

The metrology bench and corresponding data treatment were designed and built by Yves Surrel and hosted at IOGS Saint-Etienne.

9.1 Freeform test piece

We designed and manufactured a freeform test piece with the goal of challenging existing metrology methods with high altitude and slope departure from a base shape. The piece design and manufacturing, the PMD bench setup and calibration, and the measurement results are described in the article draft included below (the article was submitted and rejected a first time, a rewrite will be submitted at a further date).

Measurement of a freeform mirror with strong altitude and slope departures using deflectometry

THOMAS HOULLIER,[1,2,*] YVES SURREL,[3] THIERRY LÉPINE,[2,5] DAVID GLUCHOWSKI,[4] MARTIN BERTHEL,[4] JULIEN FOUREZ,[6] THOMAS MICHEL,[6] JEAN-PIERRE LAURET,[4] AND GUILLAUME CASSAR[1]

[1] *Sophia Engineering, 5 Rue Soutrane, 06560 Valbonne, France*
[2] *Univ-Lyon, Laboratoire Hubert Curien, UMR CNRS 5516, 18 rue Benoît Lauras, 42000 Saint-Etienne, France*
[3] *Yves Surrel Expertise & Consultancy*
[4] *Gaggione S.A.S., 3 rue de la Rolland, 01460 Montréal-la-Cluse, France*
[5] *Institut d'Optique Graduate School, 18 rue Benoît Lauras, 42000 Saint-Etienne, France*
[6] *Thales SESO, 530 rue Frédéric Joliot, 13290 Aix-en-Provence, France*
[*] *thomas.houllier@sophiaengineering.com*

Abstract: We manufactured an aluminum optical freeform mirror using slow tool servo single point diamond turning (SPDT) machining. The mirror is 50.8 mm in diameter and presents extreme freeform departures of more than 400 micrometers in altitude, 80 mrad in slope. The surface exhibits both concave and convex areas. We measured the mirror's surface appearance defects, absolute shape and high spatial frequency errors using a Phase-Measuring Deflectometry (PMD) bench. We assessed the absolute shape measurement error with the help of a complementary LUPHOScan measurement. The RMS deviation between the deflectometric and LUPHOScan absolute measurements was found to be no more than 940 nm RMS over the whole area, falling down to 55 nm RMS over the central 10 mm subaperture, confirming that PMD is a very interesting tool for freeform metrology.

1. Introduction

Recent work in the manufacturing of freeform imagers [1–4] and in individual freeform pieces [5] has demonstrated the value of single-point diamong turning (SPDT) as a fast prototyping tool for infrared imaging systems. The surface shape error as well as roughness obtained by SPDT is sufficient for IR imagers and can be used for visible wavelength imagers with further cladding and polishing.

Parallel work in freeform surface metrology allows the measurement of these freeform surfaces. We can cite null-interferometry benches [6, 7] using optical systems or Computer-Generated Hologram (CGH) as nulls, scanning multi-wavelength interferometry (*eg.* Taylor Hobson's LUPHOScan [8]), Coordinate-Measuring Machines (CMM) with optical probe [9]. For the purpose of measuring freeform surfaces, particularly in-situ, some more flexible or more accessible methods exist, such as shearing interferometry [10] and phase-measuring deflectometry (PMD) [11–14].

We designed, manufactured, and measured a freeform aluminum mirror. Our goal is to assess the capabilities of SPDT and PMD in manufacturing and measuring extreme freeform optical surfaces. The proposed shape has a plane for base shape. It is extreme in the sense that it possesses steep slopes of more than 80 mrad and maximum deviation from the plane of more than 400 µm. The proposed surface, because of its steep slopes, cannot be measured over its whole aperture using a straightforward interferometry setup, and probably neither with a CGH

null [15] (due to pattern density limits).

2. Design and manufacturing of the freeform mirror

2.1. Design

The manufactured mirror was specifically designed to be a 'torture test' both for manufacturing and metrology. The clear aperture is Φ50.8 mm. The central Φ16 mm subaperture was designed with low altitude and slope departures (less than 10 µm and less than 5 mrad respectively). The annulus from Φ16 mm to Φ46.8 mm has a maximum slope greater than 80 mrad. The outer annulus from Φ46.8 mm to Φ50.8 mm presents much higher slopes, greater than 200 mrad. Maps for the nominal mirror shape, in altitude and slope are given in Figs. 1a and 1b.

The border outside the clear aperture is a flat with four engraved cross fiducials. The total mechanical diameter is Φ60 mm. A threaded hole on the backside of the mirror is made so as to allow convenient support.

2.2. Manufacturing

The freeform mirror material is the RSA6061T6 aluminum alloy from RSP technology. The mirror (see the photos in Fig. 1c) was machined on a Moore Nanotech 350FG equipped with a Slow Slide Servo (SSS). The diamond tool, from Contour Fine Tooling, is sharpened to a 250 nm PV shape error which should produce a surface roughness lower than 10 nm.

A few hours of machining were necessary to obtain the first version of the mirror. To go from the flat blank to the final result, we needed around 20 rough machining passes, in addition to the finishing. Indeed, the surface PV is approximately 500 µm and the milled depth on a rough machining pass is around 30 µm, to ensure the tool durability. With a half hour rough pass duration, this makes a total of a dozen hours for the roughing step. The finishing step took around 20 hours. During this last step, only 4 µm were etched with a cutting speed, set by the rotational axis, of 5 m min^{-1} (40 times less than for a typical machining).

3. Deflectometry overview and experimental setup

3.1. Overview

The origin of deflectometry (not including slope filtering in the Fourier plane which is the basis of Ronchi tests [16]) dates back to more than three decades ago, emerging with various flavors in different fields such as optical metrology (along with moiré [17, 18]), mechanical engineering (with moiré [19] or without [20, 21]) or industrial inspection [22]. Today's state of the art is to use the direct reflection of sinusoidal lines patterns displayed by a computer monitor screen (Figs. 1d and 1e, showing our experimental setup). Temporal phase-stepping along x (with vertical lines) and y (with horizontal lines) is used to realize a precise acquisition of the light-emitting screen pixel coordinates, the lines phase being simply proportional to their x (vertical lines) or y (horizontal lines) coordinates. Temporal phase unwrapping [23] can be used to obtain faithfully unwrapped phase maps. In this work, we used the synthetic wavelength approach [24], allowing to map the zero phase point to the center of the display screen.

There are different levels in deflectometric measurements, differing in the amount of required calibration and data processing.

3.1.1. Qualitative

The x and y phase maps are high-pass filtered, in order to detect localized (cosmetic) defects. Indeed, the high frequency local phase variations are related to local slope variations, and displaying these will give information about mid-frequency shape defects. These maps can also be differentiated to provide maps related to local curvatures. This will enhance high frequency

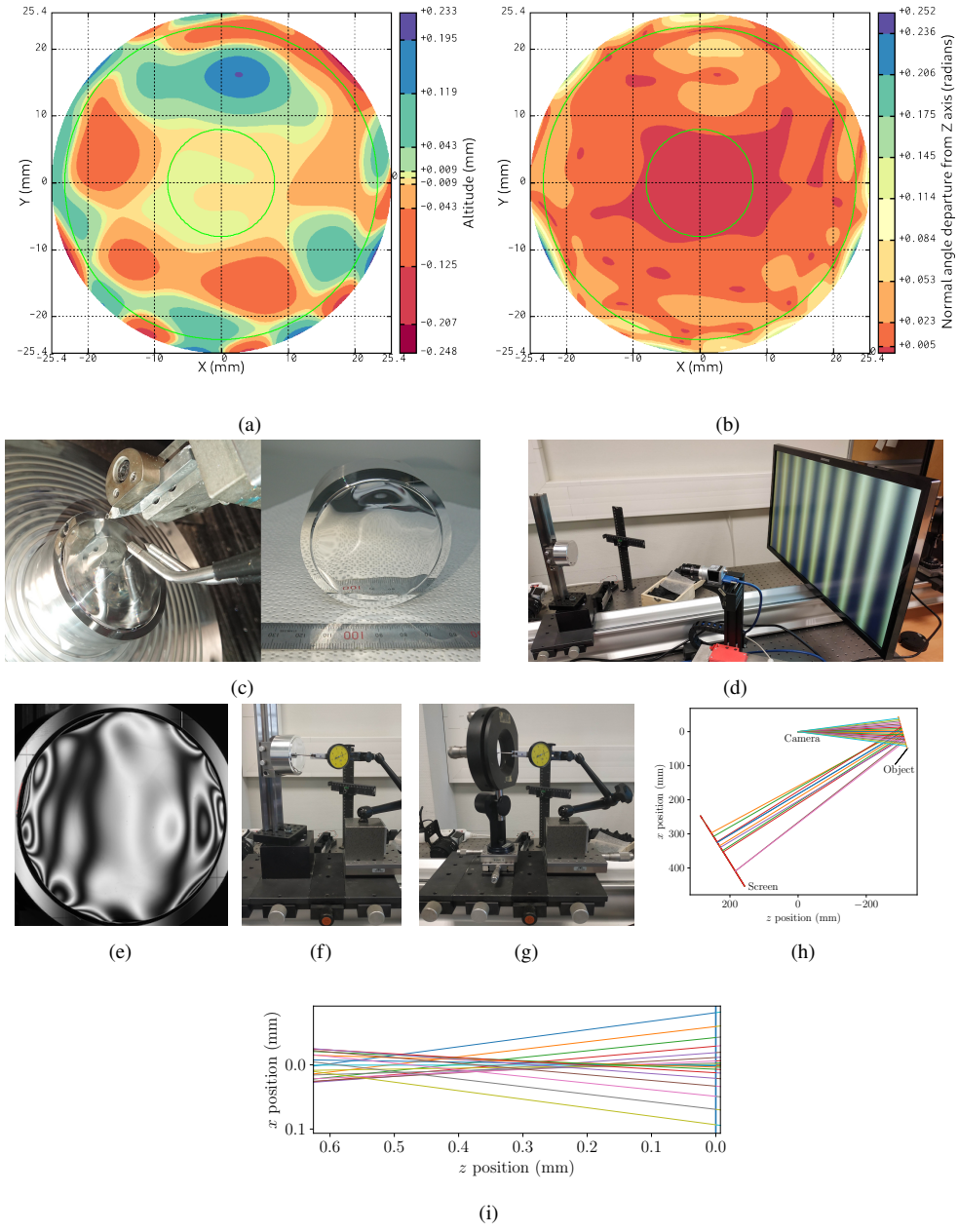


Fig. 1. (a) Mirror nominal altitude, (b) Angular departure from Z axis (slope). The central overlayed circle indicates a $\Phi 16$ mm subaperture. The second overlayed circle indicates a $\Phi 46.8$ mm subaperture. Please note the non-linear and discretized color scale for viewing purposes, the actual surface is continuous in altitude. (c) Photos of the manufactured mirror: during the manufacturing process (left) and finished (right). (d) Setup overview. (e) Reflected fringes as seen by the camera (the fiducials engraved onto the outer ring can be seen). (f) a mechanical gauge is put in contact with the freeform mirror. (g) the plane mirror is put in contact with the gauge. (h) ray trajectories in the horizontal diametral (x, z) plane in the camera frame of reference. (i) closeup view of the rays at the camera pupil, showing the lens spherical aberration.

defects detection, due to the gain of the filter corresponding to differentiation that is proportional to the spatial frequency. If only arbitrary units are sought, which can be the case in industrial applications requiring only detection based on simple thresholding criteria, no calibration nor complex data processing is necessary. If real units are needed, a rudimentary camera model (pinhole) and calibration (camera and screen position) is most often sufficient. Apart from the phase data, another very interesting outcome of deflectometric measurements is the detected amplitude (or contrast). The detected modulation amplitude *exclusively* corresponds to *reflected* light. So, scratches (but also dust, fingerprints and any surface pollution) that diffract or absorb light will lower locally this amplitude, which will make these defects clearly show up when displaying amplitude x and y derivatives. These features (simplicity, sensitivity) make this qualitative level popular in industrial inspection of parts [25].

3.1.2. Differential-quantitative

The part to be measured can be compared *in-situ* to a reference object, usually a plane mirror. In that case, only the phase deviations between the measurements on the reference and tested objects matter, making most of the systematic errors (*e.g.* lens distortion) vanish. The camera model can be restricted to a pinhole one, with or without distortion correction depending on the requirement on lateral defects positioning accuracy. Quantitative results require a moderate/good quality calibration, in which case excellent results can be obtained. For example, Huang *et al.* [26] obtain an accuracy in the nanometer range when measuring an off-axis elliptical X-ray mirror. Results on our freeform mirror are presented below. However, in our case it can be seen from the rays trajectories in Fig. 1h that the deviation from a plane is such that the reflected 'sight' rays (coming from the camera) impinge the display screen at positions that can be tens of millimiters away from what they are in the plane mirror case. In that case, residual error compensations (especially related to the position and shape of the screen) are not so well realized and thus the absolute shape reconstruction will not be as good as what is presented in [26].

3.1.3. Absolute-quantitative

Absolute shape measurement is sought. It is still a challenging issue today, as it requires ultimate precision and accuracy in the geometrical description of the whole setup: ray trajectories in object space, screen position and shape, index and thickness measurement of the screen protective window, etc., possibly requiring additional measurement tools like a CMM or laser tracker to locate the screen and camera in a single frame of reference.

3.2. Integration

To reconstruct the shape, the slopes have to be integrated. However, converting the acquired screen coordinates observed in reflection into slopes requires knowing the object shape, an egg-and-hen problem. We followed the usual iterative approach: the slopes are first evaluated assuming a plane object and then integrated to get a first shape estimate. Then, reflected screen coordinates assuming this object shape are computed. The deviations from the measured screen coordinates allow the determination of small slope corrections, that can be integrated to get a small shape correction, and the process is iterated until the shape correction falls below a given level (*e.g.* 10 nm).

3.3. Calibration

For this work, we used a 28", 3840×2160 pixel Samsung U28E570 screen, a 2592×1944 pixel Basler acA2500-14-gm camera and a 35 mm Fujinon HF35XA-5M lens opened at its smallest aperture value F/16. Different calibration steps are needed for deflectometric measurements.

- calibration of the effective screen/camera gamma curve, to ensure a purely sinusoidal profile of the fringes after display and acquisition;

- calibration of rays:
  - identification of the camera intrinsic parameters in case of a pinhole camera model;
  - identification of the distortion in case it is required;
  - identification of the ray trajectories in the object space in case of a model-free camera description, which was our choice in this work.
- identification of the screen position.

3.3.1. Effective screen/camera gamma curve

When temporal phase stepping is used, the only crucial point is related to the harmonic content of the fringe profile. To avoid applying a numerical rejection of harmonics [27], we calibrate the system response. In the case of fringes displayed by a computer screen, the best is to calibrate the screen+camera response such that the displayed fringes *appear* sinusoidal to the camera. To do this, a set of uniform linearly increasing gray levels is displayed on the screen, and the average image gray level is determined at each step, in order to have the screen/camera response (so called 'gamma' curve). This curve is then inverted to create a lookup table (LUT), thus linearizing the overall system response.

3.3.2. Identification of rays in object space

We used a model-free camera description, requiring the identification, in a fixed frame of reference, of the equation of the ray propagation line for each camera pixel. To do this, the screen is located at different distances (250 mm apart within an overall 1500 mm travel) in front of the camera which is positioned on a Newport MTM250 translation stage and moved at different distances (10 mm apart within its 250 mm range). For each screen and camera distance, the screen coordinates are acquired via a phase-stepped measurement. To prevent disturbances induced by air turbulence, a protective tunnel is placed around the bench. After determination of the translation stage movement imperfections (roll, pitch, yaw) and screen attitudes (no assumption is made on the screen being orthogonal to any axis), the rays trajectories can be determined in a common frame of reference. It can be seen in Fig. 1i that the lens we have used exhibits some amount of spherical aberration.

3.3.3. Screen position identification

With the camera set in its final geometric position in the setup (Fig. 1d), the screen position in its coordinate system can be computed using reflections off of a plane mirror at different angles. We used an approach similar to what is described in [28, 29], although our mathematical treatment seems simpler.

For each mirror position the screen *image* position (affine transform consisting in a rotation and a translation) can be determined, because the rays trajectories are known. When at least three arbitrary plane mirror positions are used, one can deduce the position of the display plane in the camera reference frame.

Let us summarize the key principles of the procedure. The camera and screen positions are fixed throughout the process. For a particular plane mirror position, the screen and its image are symmetrical with respect to the mirror plane. Moving the mirror will modify the position of this image. Let us number three mirror (and screen image) positions 1, 2 and 3. The transformation image 1 $\rightarrow$ image 2 is the composition of two successive plane symmetries: image 1 $\rightarrow$ screen and screen $\rightarrow$ image 2. It is a known mathematical result that a composition of two plane symmetries is a *pure rotation* around an axis that is the intersection of the two symmetry planes (here the mirror in positions 1 and 2 that we denote mirror 1 and mirror 2). Since the screen image positions 1 and 2 are known, we can deduce the rotation axis between them, which belongs to both mirror 1 and mirror 2. We can similarly determine an axis common to mirror 1 and mirror 3, and another one common to mirror 2 and mirror 3. As we have now two axes per mirror

position, this is enough to determine the position of the three mirror planes. From these mirror positions we can finally deduce by plane symmetry the screen position from the corresponding image positions. There is of course a redundancy as the screen position can be obtained three times from the three different mirror/image couples, allowing to check the coherence and quality of the results. Another check is to verify how close the two axes corresponding to each mirror position are; mathematically, they should intersect as they are supposed to belong to the same plane but it is of course never the case in a real measurement. We obtained a typical 200 µm distance between axes in our calibration.

Theoretically, acquiring data for more than three mirror positions and with angles and positions as far apart as possible should increase the method's accuracy as the random errors should compensate more and more. However this accuracy increase is not granted, because systematic errors remain as the screen is not perfectly flat.

4. Results

We performed a differential measurement to get the surface shape of the freeform mirror, using a plane reference mirror. We first measured the plane mirror with a Zygo GPI XP Fizeau interferometer with an absolute accuracy of $\lambda/100$ and checked it was within $\lambda/10$ flatness. In differential deflectometric measurements, it is essential to reposition a reference point of the part to be measured exactly at the same location in space as some point of the reference part to define a $z = 0$ plane. To achieve this, the following procedure was set up.

First, with the freeform mirror in measurement position, a Mitutoyo gauge 513-404-10 (minimum contact force: 0.01 N) positioned on a manual translation stage is brought just-in-contact with the mirror (Fig. 1f). An auxiliary carrier is used as a mechanical stop on the bench to be able to reposition exactly the gauge afterwards, and the gauge was removed. After measuring the freeform mirror, the gauge was replaced and the plane mirror placed on two crossed manual translation stages was then put just-in-contact with the gauge. The plane mirror can then be measured.

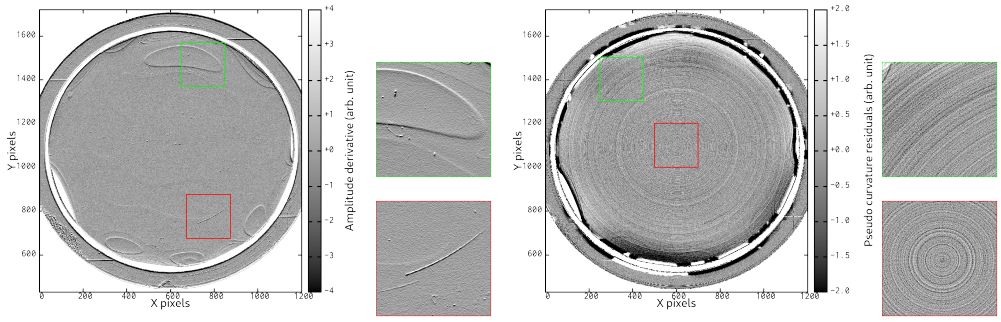
4.1. Cosmetics

The $y-$derivative of the amplitude is represented in Fig. 2a. The pseudo-curvature residuals, high-pass filtered sum of the $x-$ and $y-$derivatives of the x and y phase maps is found in Fig. 2b. The low spatial frequencies were subtracted from the pseudo curvature map by subtracting the convolution by a 31x31 pixels gaussian kernel of standard deviation 17 pixels.

These results illustrate the usefulness of deflectometry for cosmetic control of reflective surfaces, as nanometric-range defects are easily revealed with very simple data processing, despite the fact that in that case the results are only obtained in arbitrary units. In Fig. 2a, the zoomed-in view shows a slope artefact, which is confirmed to be an actual manufacturing defect in our quantitative measurement of curvatures.

4.2. Shape

The result for the shape measurement is shown in Fig. 3a. A cross-check measurement was carried out using the LUPHOscan (LS420HD) over Φ48.2 mm. The manufacturer claims an accuracy of ±50 nm (2σ) for this instrument. We performed a comparison between the deflectometry, LUPHOScan and nominal altitudes over Φ45 mm to crop out edge effects in the measurements. The piston, tip, tilt and clocking were conjointly optimized between the three maps to minimize the RMS altitude error. Some amount of lateral magnification (0.3 %) was also introduced. As the deflectometric results crucially depend on the system geometry identification, and particularly on the screen pixel size, an error of this order of magnitude is not impossible, as this feature is not easy to measure with an extreme precision. The freeform shape is very complex, so it seems unlikely that the improvement introduced by this magnification correction be only coincidental.



(a) (b)

Fig. 2. (a) $y-$derivative of amplitude, in arbitrary units. The measurement highlights qualitatively the scratches and defects on the mirror surface. No filtering of the measurement data was applied. (b) Pseudo curvature defects, high-pass filtering of the sum of the phases derivatives.

A horizontal cross-section of the cross-differences as well as altitude measurement is shown in Fig. 3e. The RMS error between the LUPHOScan and deflectometry maps as a function of subaperture size is plotted in Fig. 3f (the RMS altitude of the piece as manufactured is given as indicative of the difficulty of the measurement on the same plot).

LUPHOScan and PMD measurements agree better with one another than with the nominal map (see Fig. 3d). This confirms that the piece was manufactured with significant errors. This is unsurprising given the difficulty of this manufacturing task. This gives data on what level of accuracy one should expect when manufacturing a very difficult freeform surface in SPDT, on the first try, without subsequent polishing/correcting steps.

Taking the LUPHOScan measurement as ground truth, we can assess the accuracy of our deflectometry results with regards to absolute surface shape. Difference values have to be taken with caution, and are certainly an *upper bound* of the deviation. The problem comes from the pixel-wise subtraction. Artificially large errors can come from a slight discrepancy in position and orientation between the two sets of values. The mirror has slopes over 0.1 rad, corresponding to 1 µm per 10 µm lateral displacement. This shows that any slight discrepancy between the pixel positions during the difference may end up with large artificial errors. The altitude error between deflectometry and LUPHOScan is shown on Fig. 3c. We see that the error is smaller around the center of the piece and larger near the outer edge, which presents the largest freeform departures, both in altitude and slopes. Thus, the deflectometry measurement error seems to be related to the freeform departure, as shown in Fig. 3f. On a smaller subaperture of 10 mm, over which the freeform departure is 4 µm RMS, we obtain a measurement error of 55 nm RMS. This is a sufficient surface accuracy for low-end imaging applications, especially in IR. This is also sufficient for most lighting applications. On the other hand, once the whole 45 mm comparison subaperture is considered, which includes the steep outer parts of the piece and a 47 µm RMS freeform departure, the error of our proposed deflectometry method is at most 937 nm RMS.

4.3. High/medium frequency machining defects

From the shape measurement, it is also possible to extract the medium- and high-frequency component to get some relevant and quantitative machining defects. As can be seen in Fig. 3b, an axisymmetric defect is clearly visible, the high-frequency part of which was already visible in the cosmetic result in Fig. 2b. It corresponds to the linear translation defect of the machining tip. Due to the axisymmetry, it is possible to perform an azimutal filtering to get the smoothed

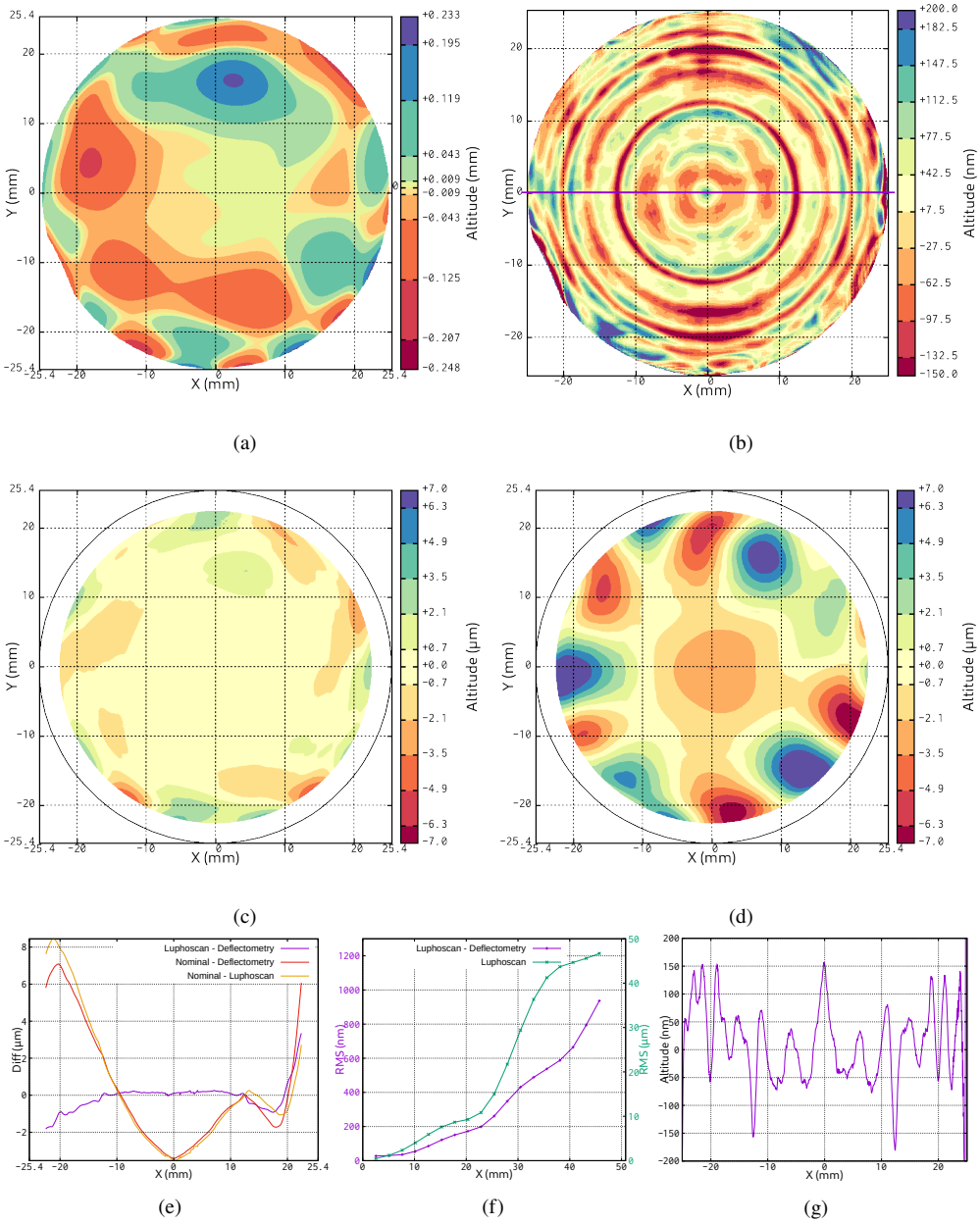


Fig. 3. (a) Deflectometry measurement map. (b) High frequency machining defects. A best fit XY polynomial of degree 20 was subtracted from the measured altitude. The scale is cropped in the range [-150;200]nm. The horizontal line indicates the cross-section position in (g). (c) Difference deflectometry/LUPHOScan. Note the scale is cropped at [-7,7] µm. The black circle indicates the size of the aperture that was measured with the deflectometric method. (d) Difference LUPHOScan/nominal. Same scale as in (c). (e) Horizontal cross-section across the center of the piece: all crossed differences. (f) RMS error deflectometry/LUPHOScan as a function of considered subaperture diameter (left axis). RMS of the manufactured surface (right axis). (g) Cross-section of the smoothed axisymmetric machining medium/high frequency defect.

machining defect, of which a cross-section is shown in Fig. 3g.

5. Conclusion

We have shown what could be expected from blind (without measurement and correction) SPDT machining for a freeform piece with extreme departures (>47 µm RMS and slopes of more than 80 mrad). The machining error is approximately 17 µm PV and 3 µm RMS (see Fig. 3d). Additionaly, we have assessed the measurement error of a simple, cost-effective and fast deflectometric method for shape measurement: 55 nm RMS for areas with low-departure freeforms (4 µm RMS) and 937 nm RMS once extreme freeforms are considered (47 µm RMS). The values can be read on Fig. 3f. While insufficent for testing strongly freeform optical surfaces for imaging applications, the simplicity of the method makes it a good lead (with some accuracy improvements) for online testing during the first machining passes, since the manufacturing error of a blind process is still greater than those of our proposed method, where absolute shape is concerned.

In accordance with the well-known relevancy of deflectometry for local shape defects detection, we have shown minute scratches that were not detected with the LUPHOScan, and could not have been due to its insufficient spatial resolution. Quantitatively, the high spatial frequency content can be extracted from the shape measurement. High frequency machining defects can be measured with near-nanometric resolution. This is useful to assess the surface error that can be attributed to mid-spatial frequencies.

Certainly, a deflectometry bench should be a very useful addition to optical testing equipments suitable to freeform optics. As compared with existing techniques (CMM and/or laser trackers, scanning multi-wavelength interferometry, sophisticated nulls or CGH for interferometers), deflectometry has the advantages of:

- cost-effectiveness (overall cost of some thousands of dollars instead of some hundreds of thousands),
- measurement speed (tens of seconds instead of tens of minutes),
- spatial resolution (the usual advantage of full-field techniques against scanning ones),
- dynamic range (field of view —at least on quasi-planar or concave surfaces, slopes),
- extremely simple data processing for cosmetic defects assessment.

The well known weakness of deflectometry is the difficulty to access the low frequency shape information. However, the presented results show that even on a very challenging freeform surface, provided a convenient calibration and data processing has been implemented, the shape measurement accuracy begins to convincingly approach what can be obtained with the best instruments on the market.

Planned improvements of the deflectometric setup are:

- improvement of the screen pixel size measurement;
- during the calibration step, in situ shape measurement of the monitor screen by fringe projection;
- during the calibration step and for each measurement, correction of phase maps by taking into account the heterogeneous point spread function of the system (camera lens + object under test + screen being out of focus).

Funding. Association Nationale de la Recherche et de la Technologie.

Acknowledgments. We thank Simon Henrot, working at Thales Alenia Space for his contributions towards the data analysis in the present work. We also thank Jean-Philippe Sneed and Hervé Benard, also from Thales Alenia Space, for their time and effort. This work was financed by ANRT as part of a CIFRE doctoral research program between Laboratoire Hubert Curien, Sophia Engineering and Thales Alenia Space.

Disclosures. The authors declare no conflicts of interest.

Data availability. Data underlying the results presented in this paper are not publicly available at this time but may be obtained from the authors upon reasonable request. The freeform torture test mirror shape definition is available in the Supplemental Document.

Supplemental document. See Supplement 1 for supporting content.

References

1. Z. Shen, J. Yu, Z. Song, L. Chen, Q. Yuan, Z. Gao, S. Pei, B. Liu, and J. Ye, "Customized design and efficient fabrication of two freeform aluminum mirrors by single point diamond turning technique," Appl. Opt. **58**, 2269–2276 (2019).
2. X. Zhang, H. Gao, Y. Guo, and G. Zhang, "Machining of optical freeform prisms by rotating tools turning," CIRP Annals **61**, 519 – 522 (2012).
3. Z. Li, F. Fang, J. Chen, and X. Zhang, "Machining approach of freeform optics on infrared materials via ultra-precision turning," Opt. Express **25**, 2051–2062 (2017).
4. M. Beier, J. Hartung, T. Peschel, C. Damm, A. Gebhardt, S. Scheiding, D. Stumpf, U. D. Zeitner, S. Risse, R. Eberhardt, and A. Tünnermann, "Development, fabrication, and testing of an anamorphic imaging snap-together freeform telescope," Appl. Opt. **54**, 3530–3542 (2015).
5. Z. Li, F. Fang, X. Zhang, X. Liu, and H. Gao, "Highly efficient machining of non-circular freeform optics using fast tool servo assisted ultra-precision turning," Opt. Express **25**, 25243–25256 (2017).
6. K. Fuerschbach, K. P. Thompson, and J. P. Rolland, "Interferometric measurement of a concave, φ-polynomial, zernike mirror," Opt. Lett. **39**, 18–21 (2014).
7. H. P. Stahl, C. Alongi, A. Arneson, R. Bernier, B. Brown, D. Chaney, G. Cole, J. Daniel, L. Dettmann, R. Eng *et al.*, "Survey of interferometric techniques used to test jwst optical components," in *Interferometry XV: Techniques and Analysis,* vol. 7790 (International Society for Optics and Photonics, 2010), p. 779002.
8. "LUPHOScan Non-contact Profilometers," https://www.taylor-hobson.com/products/non-contact-3d-optical-profilers/luphos. (Accessed: 07 February 2020).
9. R. Henselmans, L. Cacace, G. Kramer, P. Rosielle, and M. Steinbuch, "The nanomefos non-contact measurement machine for freeform optics," Precis. Eng. **35**, 607 – 624 (2011).
10. Y.-S. Ghim, H.-G. Rhee, A. Davies, H.-S. Yang, and Y.-W. Lee, "3d surface mapping of freeform optics using wavelength scanning lateral shearing interferometry," Opt. Express **22**, 5098–5105 (2014).
11. I. Trumper, B. T. Jannuzi, and D. W. Kim, "Emerging technology for astronomical optics metrology," Opt. Lasers Eng. **104**, 22 – 31 (2018). Optical Tools for Metrology, Imaging and Diagnostics.
12. G. Butel, "Analysis and new developments towards reliable and portable measurements in deflectometry," (2013).
13. G. Häusler, C. Faber, E. Olesch, and S. Ettl, "Deflectometry vs. interferometry," in *Optical Measurement Systems for Industrial Inspection VIII,* vol. 8788 P. H. Lehmann, W. Osten, and A. Albertazzi, eds., International Society for Optics and Photonics (SPIE, 2013), pp. 367 – 377.
14. A. V. Maldonado, P. Su, and J. H. Burge, "Development of a portable deflectometry system for high spatial resolution surface measurements," Appl. Opt. **53**, 4023–4032 (2014).
15. C. Zhao and J. H. Burge, "Optical testing with computer generated holograms: comprehensive error analysis," in *Optical Manufacturing and Testing X,* vol. 8838 O. W. Fähnle, R. Williamson, and D. W. Kim, eds., International Society for Optics and Photonics (SPIE, 2013), pp. 119 – 130.
16. V. Ronchi, "La prova dei sistemi ottici," (1925).
17. O. Kafri, "Noncoherent method for mapping phase objects," Opt. Lett. **5**, 555–557 (1980). Publisher: Optical Society of America.
18. O. Kafri and I. Glatt, "Moire Deflectometry: A Ray Deflection Approach To Optical Testing," Opt. Eng. **24**, 246944 (1985). Publisher: International Society for Optics and Photonics.
19. F. Ligtenberg, "The Moiré Method: A New Experimental Method for the Determination of Moments in Small Slab Models," in *Proc. SESA,* vol. 12 (1954), pp. 83–98.
20. R. Ritter and R. Hahn, "Contribution to analysis of the reflection grating method," Opt. Lasers Eng. **4**, 13–24 (1983).
21. Y. Surrel, N. Fournier, M. Grédiac, and P.-A. Paris, "Phase-stepped deflectometry applied to shape measurement of bent plates," Exp. Mech. **39**, 66–70 (1999).
22. Y. Y. Hung, G. C. Jin, and S. H. Tang, "Surface Inspection Of Automotive Bodies By Reflective Computer Vision," in *Industrial Laser Interferometry II,* vol. 0955 (International Society for Optics and Photonics, 1988), pp. 37–40.
23. C. Zuo, L. Huang, M. Zhang, Q. Chen, and A. Asundi, "Temporal phase unwrapping algorithms for fringe projection profilometry: A comparative review," Opt. Lasers Eng. **85**, 84–103 (2016).
24. J. C. Wyant, "Testing Aspherics Using Two-Wavelength Holography," Appl. Opt. **10**, 2113–2118 (1971). Publisher: Optical Society of America.
25. S. Kammel and F. Puente Leon, "Deflectometric Measurement of Specular Surfaces," IEEE Transactions on Instrumentation Meas. **57**, 763–769 (2008). Conference Name: IEEE Transactions on Instrumentation and Measurement.
26. R. Huang, P. Su, J. H. Burge, L. Huang, and M. Idir, "High-accuracy aspheric x-ray mirror metrology using Software Configurable Optical Test System/deflectometry," Opt. Eng. **54**, 084103 (2015).
27. Y. Surrel, "Design of algorithms for phase measurements by the use of phase stepping," Appl. Opt. **35**, 51–60 (1996).

28. T. Zhou, K. Chen, H. Wei, and Y. Li, "Improved system calibration for specular surface measurement by using reflections from a plane mirror," Appl. Opt. **55**, 7018–7028 (2016).
29. Y.-L. Xiao, S. Li, Q. Zhang, J. Zhong, X. Su, and Z. You, "Optical fringe-reflection deflectometry with bundle adjustment," Opt. Lasers Eng. **105**, 132 – 140 (2018).

Measurement of a freeform mirror with strong altitude and slope departures using deflectometry: supplemental document

We compile here additional data about the freeform mirror, including its complete freeform surface definition for reproducibility. We also include complementary information about our setup, calibration and results.

1. ADDITIONAL MAPS OF NOMINAL SHAPE

Found here are additional maps for the nominal shape of the freeform mirror, these can help estimate the measurability.

- X and Y first derivatives: Figs. S1a, S1b
- X and Y curvatures: Figs. S1c, S1d

2. FREEFORM SURFACE DEFINITION

The complete mathematical definition for the nominal freeform mirror surface shape is as follows:

- Clear aperture semi-diameter: 25.4 mm
- Base shape: Plane
- Freeform type: *Zernike Standard Sag* (OpticStudio)
- Normalization radius: 26.8 mm
- Maximum freeform coefficient: Z78
- Freeform coefficients: see Fig. S2

The piston, tip, tilt and focus terms, as well as all rotationally symmetric terms were set to zero. The normalization radius for the Zernike polynomials is 26.8 mm, which is a bit larger than the clear aperture of the mirror (25.4 mm in semi-diameter) so as to avoid including the very high slopes that are likely to appear on the edge of Zernike polynomial terms.

In order to reproduce the surface, one can input the Tab. S2 coefficients in OpticStudio, then, taking the usual OpticStudio coordinate system, perform a symmetry around the Y axis and finally flip the Z axis sign. All these transformations are the result of choices made for the sake of convenience for working using the different coordinate systems between optical design software and manufacturing software.

3. INTERFEROMETER MEASUREMENT ATTEMPT AND PLANE REFERENCE MIRROR

To show how difficult interferometric measurements would be, we put the freeform mirror in front of a Zygo Fizeau interferometer with a reference plane. We tried measuring the Φ16 mm subaperture. Fig. S3a shows the Zygo in alignment mode. A single mirror position cannot be used to measure the whole Φ16 mm subaperture, only very partial measurements were obtained, not enough for a cross-validation.

Additionally, we provide the interferometric measurement of the plane mirror that was used as plane reference for the screen position identification and for the quantitative deflectometric shape measurement (Fig. S3b).

4. IDENTIFICATION OF RAYS IN OBJECT SPACE

The frame of reference orientation follows the OpenGL convention and is visible in Fig. S1e. Additionally, we give the gamma curve we identified during our screen/camera calibration: Fig. S1f.

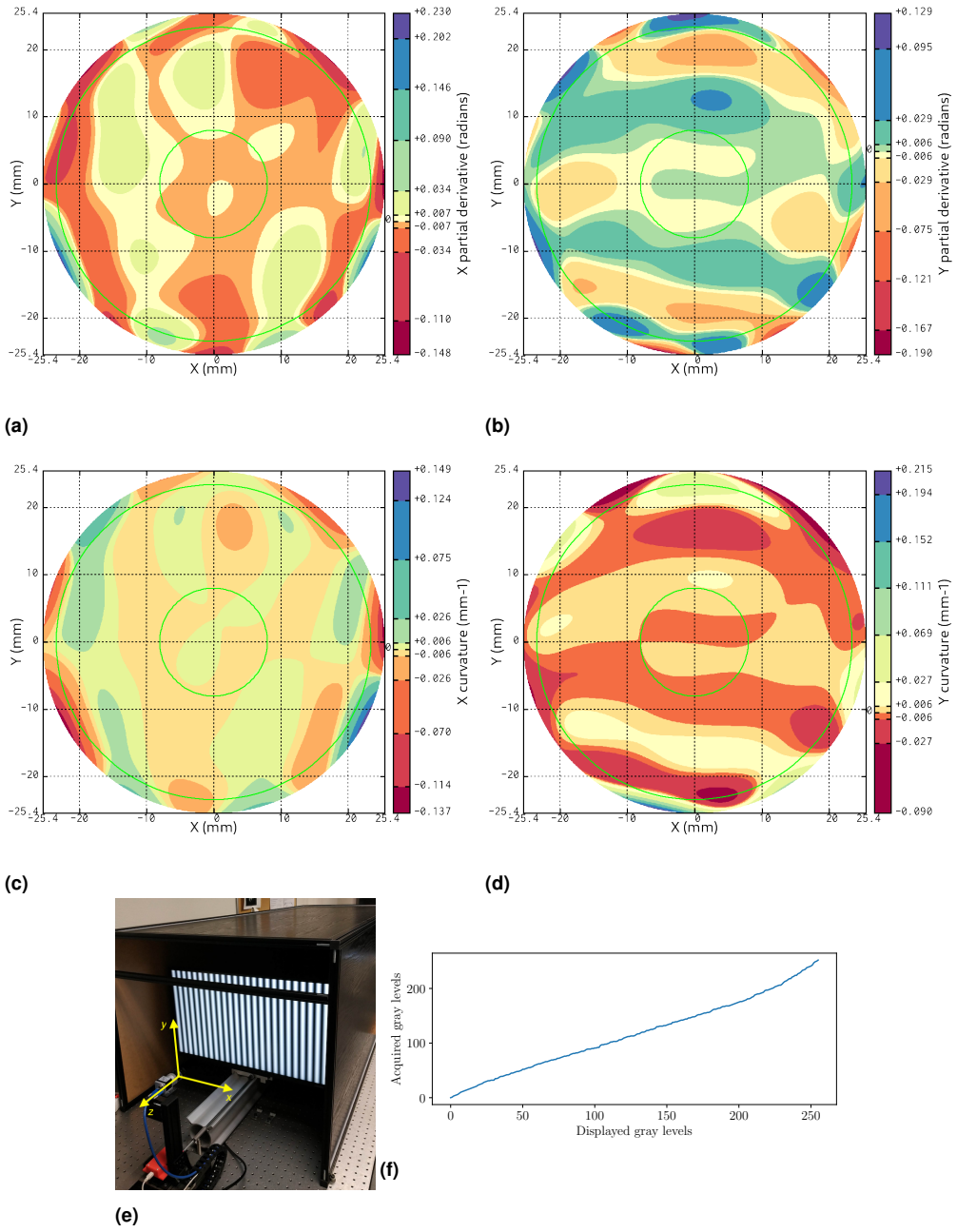
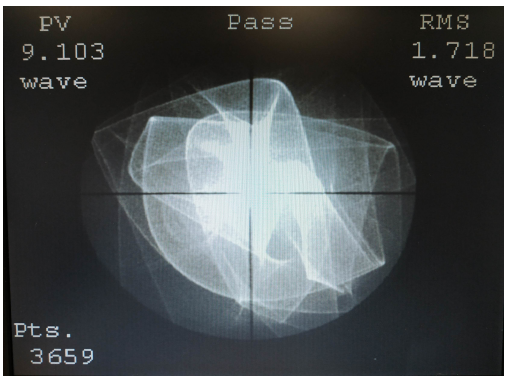


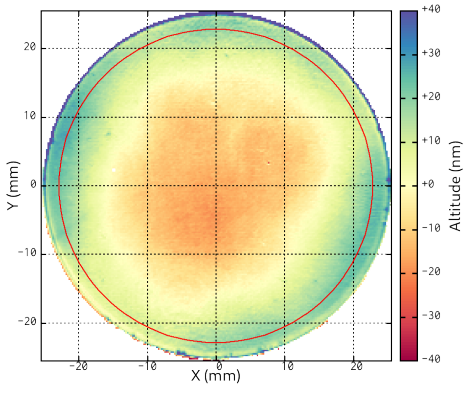
Fig. S1. (a) Nominal shape: X partial derivatives. (b) Nominal shape: Y partial derivatives. (c) Nominal shape: X curvature. (d) Nominal shape: Y curvature. (e) Calibration bench with its tunnel for protection against air turbulence. (f) Identified gamma curve of the screen+camera set.

Fig. S2. Zernike coefficients values used for manufacturing

i	n	m	value (mm)	i	n	m	value (mm)	i	n	m	value (mm)
1	0	0	0.0	27	6	-6	-9.686649E-003	53	9	-7	9.896096E-003
2	1	1	0.0	28	6	6	8.278042E-003	54	9	9	3.967977E-002
3	1	-1	0.0	29	7	-1	-9.270868E-003	55	9	-9	-6.526665E-003
4	2	0	0.0	30	7	1	-7.406787E-003	56	10	0	0.0
5	2	-2	2.185503E-002	31	7	-3	6.692923E-003	57	10	-2	4.134598 E-003
6	2	2	1.069417E-002	32	7	3	-3.025685E-003	58	10	2	2.648275E- 003
7	3	-1	2.335031E-002	33	7	-5	-1.444321E-002	59	10	-4	6.46039 5E-003
8	3	1	8.194023E-003	34	7	5	-2.742256E-003	60	10	4	9.338736E- 004
9	3	-3	6.600191E-004	35	7	-7	4.318112E-003	61	10	-6	-1.20788 9E-002
10	3	3	-1.123816E-002	36	7	7	2.050808E-002	62	10	6	6.996871E -003
11	4	0	0.0	37	8	0	0.0	63	10	-8	2.166373E-003
12	4	2	-4.554537E-003	38	8	2	1.336044E-002	64	10	8	9.027355E -004
13	4	-2	2.335059E-002	39	8	-2	8.385570E-003	65	10	-10	-1.934 656E-002
14	4	4	-9.342982E-003	40	8	4	-1.655579E-003	66	10	10	3.91125 6E-003
15	4	-4	1.591519E-002	41	8	-4	1.534804E-002	67	11	-1	1.12497 6E-002
16	5	1	2.278314E-003	42	8	6	2.679274E-003	68	11	1	7.732164E- 004
17	5	-1	1.297572E-003	43	8	-6	3.929712E-003	69	11	-3	-5.9869 39E-003
18	5	3	-3.363955E-003	44	8	8	-1.095440E-003	70	11	3	-1.69165 2E-003
19	5	-3	-1.740357E-002	45	8	-8	4.171808E-005	71	11	-5	3.7897 61E-003
20	5	5	3.816153E-003	46	9	1	-3.317181E-003	72	11	5	4.427385E -003
21	5	-5	-6.307519E-003	47	9	-1	1.204005E-002	73	11	-7	7.1307 28E-003
22	6	0	0.0	48	9	3	-3.799028E-003	74	11	7	7.524236E-003
23	6	-2	1.282647E-002	49	9	-3	-1.229129E-003	75	11	-9	-2.161 745E-003
24	6	2	1.196439E-002	50	9	5	6.843080E-003	76	11	9	1.764461E- 002
25	6	-4	9.213908E-003	51	9	-5	-1.111872E-004	77	11	-11	-4.04 0449E-003
26	6	4	-5.034138E-003	52	9	7	9.013520E-003	78	11	11	-1.79366 8E-002



(a)



(b)

Fig. S3. (a) Zygo Fizeau interferometer in alignment mode. This is the best alignment we could obtain. The high values of freeform departures make alignment difficult. (b) Reference plane mirror interferometric measurement. The altitude PV within the red circle is about 40 nm.

9.2 Surface defects measurements

Additionally, we performed deflectometric qualitative surface defects measurements on other samples with a great variety in general characteristics:

- The 2 inches diameter plane mirror used as reference in the freeform piece shape measurement.
- A Φ250 mm spherical BK7 reference (raw glass, only partially specular), concave of radius of curvature 685.88 mm (Fig. 9.1).
- A ≈Φ44 mm asphero-toroidal mirror, concave with a base shape of approximate radii of curvature 140 mm and 68 mm (Fig. 9.2). The material is titanium.

We perform the exact same measurement referred to as *qualitative* in the above article draft. It allows the detection of surface defects and estimating their position and spatial extent on the sample. No geometrical setup calibration is required in order to obtain the included measurements. Overall, this method is a cheap and convenient way to perform qualitative appearance defect measurement. This can be the basis of surface inspection in the sense of ISO 10110-7 (size and number of defects) [263, 264]. The depth of the defects is not accessible without further geometrical calibration (as explained for the freeform test piece shape measurement).

Fig. 9.3 shows the defects on the surface of the reference plane mirror. We can see numerous digs and scratches[1] as well as possibly some dust particles. The mirror is seen with a slight perspective deformation as the camera is not directly in front of the mirror.

Fig. 9.4 shows the defects on the surface of the reference sphere. The reflection on the back face was neutralised using black nail polish and black adhesive. Scratches and digs are visible. The circular hole in the data at the bottom left corresponds to a sticker that would allow clocking angle measurement of the surface. The surface coverage of the measurement is only partial. A complete coverage could be reached using two acquisitions or alternatively by altering the setup geometry so that the camera field through the mirror is comprised within the display boundaries.

Fig. 9.5 shows the defects on the asphero-toroidal mirror. A few digs are visible and some dust. The surface is comparatively of better quality than for the other samples. As was the case for the spherical mirror, the measured aperture is cropped, here at the bottom, left and right of the surface. We could likewise alter the setup geometry to obtain a full aperture measurement.

We have shown how, with only minimal equipment, many different measurements could be performed: cosmetic, curvatures, absolute shape, mid-spatial frequency content. Additionally, we have exhibited measurements on a freeform

[1] Probably the result of years of handling by IOGS students.

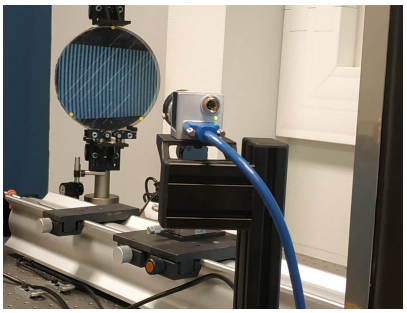


Figure 9.1: Photo of the spherical BK7 reference installed in the deflectometric bench.

Figure 9.2: Photo of the toroidal mirror.

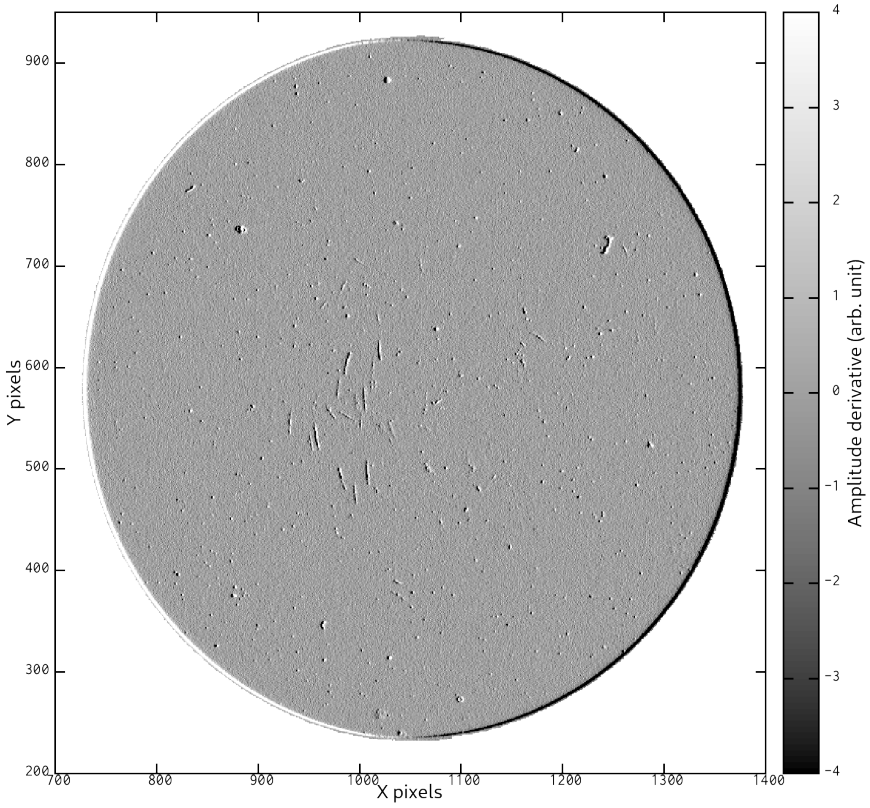


Figure 9.3: Plane mirror: X amplitude derivative, arbitrary units.

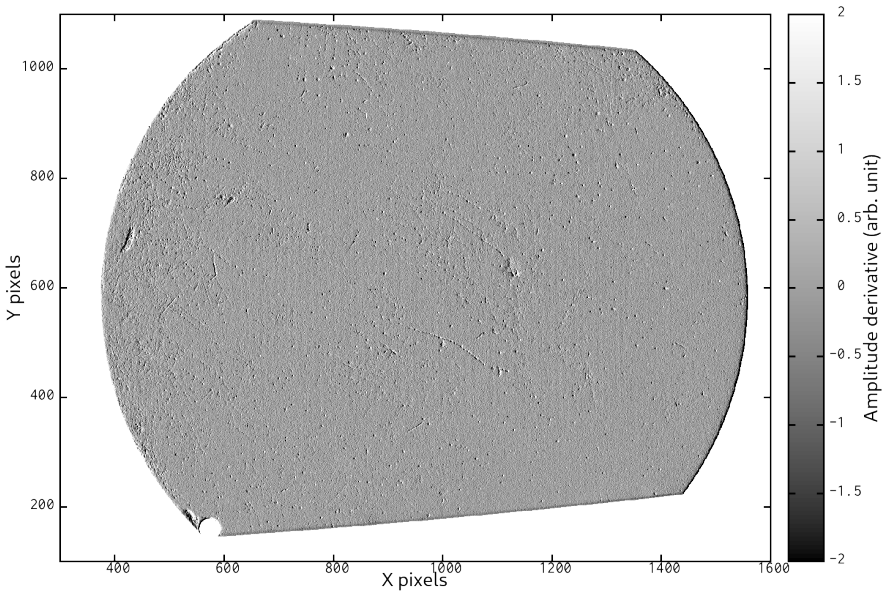


Figure 9.4: Spherical mirror: X amplitude derivative, arbitrary units.

benchmark test which would have been impossible using most conventional measurement instruments and quantified the error in absolute shape. While the numerical result of nearly 1 micron RMS in absolute shape error is rather large for many applications (including most imaging), we think this value is caused mainly by the extreme slopes of our sample. We think further research on the topic of surface shape measurement for extreme freeform samples would be worthwhile. It would unlock further freedom for the optical designers to include extreme diopters in their design.

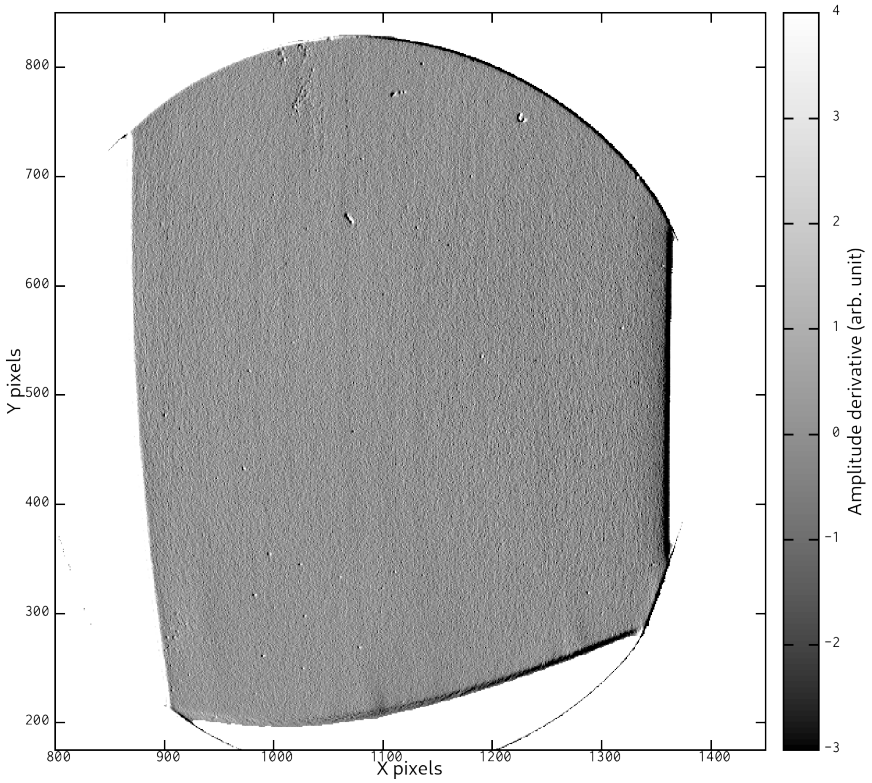


Figure 9.5: Asphero-toroidal mirror: X amplitude derivative, arbitrary units.

Chapter 10

Conclusion

Over the course of the present work, we have brushed many topics related to the design and metrology of freeform optics for imaging. Namely, we gave a short historical and technical context for the use of freeform optics in telescope design (Chapter 1). We made detailed reminders on the various current freeform representations and related mathematical tools (Chapter 2). We also documented what we considered to be the conventional way of carrying out the optical design of unobscured telescopes using the software tools available to the community (Chapter 3). We have shown the result of a study on a TMA and provided all the data to replicate our work (Chapter 4). On the metrology side of things, we have provided a short review of metrology methods that seem the most promising when it comes to measuring the surface shape of freeform optics, including with high departures in altitude and slope (Chapter 8). While the part of this memoir cited above presents only limited academic novelty, what we hope to have accomplished with these reminders is to have made available a self-standing introduction to telescope and imaging systems design up-to-date with freeform optics.

Three topics were the objects of more detailed work and gave rise to hopefully interesting contributions:

- Sequential raytracing implementation (Chapters 5 and 6)
- Search algorithms for optical design (Chapter 7)
- Phase-Measuring Deflectometry for the metrology of extreme freeform surfaces (Chapter 9)

Sequential raytracing for the simulation and design of imaging systems was summed up in all its basic principles. All the required information was given for an actual implementation of raytracing routines. We have developed a raytracing program in Common Lisp with sufficient performance so as to make it actually usable in real design problem cases[1] (and it was indeed used in the very

[1] Sadly this program is not available to the public, even upon request, in spite of its author's good will in this regard.

resource-intensive study in [144]). Conversations that we had on the occasion of various conferences, meetings and workshops impressed on us the importance of ray-aiming and the frustration it generated among many other optical designers with a programming streak to their skill set. The lack of published work on the matter prompted us to lay out the problem clearly and propose ideas useful for solving it. We do not know exactly what the solution implemented in commercial software is, nor whether it is the same across different software, but the programmers were undoubtedly smarter than we have been in solving ray-aiming, as their solution is more robust, even when it comes to (most) tricky freeform cases, and an order of magnitude faster than what we tried. We hope the documentation we produced on the subject will serve as a stepping stone for would-be raytracing programmers.

With regard to search algorithms applied to optical design, we demonstrated in our works [144, 193] the advantages there are in keeping up with the state of the art from the applied mathematics disciplines. We feel the optical design community, in spite of its historical involvement in automatic design, is lagging behind when it comes to tools and design methods that are nowadays widespread in many other engineering disciplines. The silver lining in the current state of affairs is that there are many low-hanging fruits available for interdisciplinary research work between optical design and applied mathematics. We have given some of these topics in Section 7.4, but this is just the tip of the iceberg. The researchers in the applied mathematics field we have had exchanges with were all anxious to find a domain of application for their work, especially one so untouched as optical design is (comparatively to, say, structural design in civil engineering). In our opinion, and given our work on the topic, optical design does not present any fundamental difficulty that would make it harder to take advantage of tools that are already in use in automatic design frameworks across almost all other engineering disciplines. Given the usefulness for optical design of automatic design tools, especially as freeform optics is more and more prevalent, the fact that these tools, in themselves, are already developed and widespread and that there are experts anxious to help the optical design community apply them, we may but exhort researchers and institutions to allocate more research effort in this direction.

The work carried out on the measurement of a freeform torture test using PMD demonstrated the versatility of deflectometry in obtaining measurements (albeit not very accurate yet) on surfaces with locally strongly concave and convex areas. The freeform departures we worked with are far beyond what is currently considered, even in hypothetical optical designs. Our goal was to push the envelope in terms of acceptable freeform altitude and slope departures. Our hope is that one day optical designers will be able to design such extreme pieces with the confidence that they are manufacturable. This would unlock even greater freedom for optical designers: they could start building high performance designs with surface shapes that are considered ridiculous as of today. If completely general (with regards to shape) manufacturing and metrology methods were to become widely available in optical shops, then the manufacturing of freeform optics could perhaps become as easy as that of axi-symmetric

optics is today[2].

Further work There are several subjects that can researched further and can give rise to quickly obtained and interesting results. We plan to either investigate these matters ourselves with the help of our colleagues or launch other research programs.

- We think Clenshaw-Favard theorem has potential as a freeform polynomials generating tool (Section 2.1.3.3). Using this kind of *hyperparametrized* freeform polynomials representation in optical design could perhaps allow: 1. Smoothly transitioning from one freeform polynomial to another (Zernike to XY for exemple) during automatic design. 2. Generating polynomials seldom used in optics but no less interesting. 3. Unifying (most!) freeform polynomials representations in software under a single implementation, which, though general, would be quite fast due to Clenshaw.
- Multi-objective, constrained, surrogate or Pareto optimization frameworks are all very accessible and worthwhile investigations using existing optical design software tools.
- We have begun work with our fellow researcher Maxim Sakharov to apply his advanced black box optimization methods to more complex optical systems.
- We have begun work with another colleague to optimize optical systems using a radically analytical approach[3].
- Yves Surrel is improving his deflectometry bench and data treatment. With adequate measurement procedure and subsequent treatment, he is hoping to obtain measurements of absolute surface shapes with an accuracy better by at least an order of magnitude compared to what we have shown in the present work.

[2] Freeform or not: not that easy when the error budget is in the nanometers!
[3] We do not want to give too much details until an article is published.

Published works

We give here a list of the works that were published in relation with the present thesis.

Articles

Peer-reviewed articles.

- Thomas Houllier and Thierry Lépine. "Comparing optimization algorithms for conventional and freeform optical design". In: *Opt. Express* 27.13 (June 2019), pp. 18940–18957. DOI: 10.1364/OE.27.018940
- Thomas Houllier, Yves Surrel, Thierry Lépine, David Gluchowski, Martin Berthel, Jean-Pierre Lauret, and Guillaume Cassar. "Measurement of a freeform mirror with strong altitude and slope departures using deflectometry". In: (2021). Sent to a journal and rejected. In the process of being rewritten. A draft of this article is included in the body of the present manuscript.

Conference Proceedings

- Thomas Houllier, Nicolas Rousselet, Yves Surrel, and Thierry Lépine. "Advanced optical freeform substrates fabricated by ceramic 3D printing and controlled by deflectometry". In: *Optical Fabrication, Testing, and Metrology VI*. Ed. by Sven Schröder and Roland Geyl. Vol. 10692. International Society for Optics and Photonics. SPIE, 2018, pp. 171–182. DOI: 10.1117/12.2312649
- Maxim Sakharov, Thomas Houllier, and Thierry Lépine. "Mind Evolutionary Computation Co-algorithm for Optimizing Optical Systems". In: *Proceedings of the Fourth International Scientific Conference "Intelligent Information Technologies for Industry" (IITI'19)*. Ed. by Sergey Kovalev, Valery Tarassov, Vaclav Snasel, and Andrey Sukhanov. Cham: Springer International Publishing, 2020, pp. 476–486. ISBN: 978-3-030-50097-9. DOI: 10.1007/978-3-030-50097-9_48

Presentations & Miscellaneous

- Thomas Houllier and Thierry Lépine. "Search algorithms and optical systems design". In: Paris: Zemax ENVISION2019 (private non-indexed conference), Mar. 2019. DOI: 10.13140/RG.2.2.31951.12969
- Thomas Houllier and Thierry Lépine. "Optical design and optimization strategies: state of the art and future developments". In: Nice, France: Workshop on Machine-Learning-Assisted Image Formation, July 2019. DOI: 10.13140/RG.2.2.28749.87524

Acknowledgments and Thanks

First and foremost I would like to thank Thierry Lepine, who I was glad to have as thesis advisor. Even though I worked in Sophia Antipolis and he in Saint-Etienne, Thierry was always attentive to my progress and the communication has always been frequent, excellent and easy. He offered me many occasions to promote and publish my work and seemed to always put his students first. Together, we navigated these three years of doctoral research guided by his sure yet amicable wisdom.

This research work was financed by the private engineering company Sophia Engineering, where I worked during my research and where I am still working, now as an engineer. The idea of the founders Vincent David (CEO) and Laurent Escarrat (CTO) to initiate doctoral research in their company was an interesting surprise to myself and to many researchers I came across. The environment they created is pleasant to the research-oriented people that optical engineers often are, either through personal inclination or by virtue of the nature of the discipline.

I thank my colleagues of the optics department at Sophia Engineering. Guillaume Cassar, as Sophia's advisor for the PhD, was always there to ensure our research plans were viable and that we could stick to the planning. He made his expertise and vast experience in optics available whenever we needed it. Mathieu Guilhem deserves a mention for being a fountain of technical creativity, together with him and Guillaume, we exchanged many ideas as to the directions we could take the research in. I also thank Mathieu for bearing his frequent role as a most insightful sounding board. I thank my esteemed colleagues in the optical design trade: Jacques Dalmau and Laurie Capobianco-Peigne.

Yves Surrel (Yves Surrel Expertise & Consultancy) made his phase-measuring deflectometry bench, software and his personal time available to the metrology experiment we performed in the course of the present work. He is the main contributor by far in the deflectometry part of this thesis and I acted as coordinator between the many entities that were involved in the work.

My colleagues in the CNES R&T FAME were Vincent Costes (CNES) and Nicolas Rousselet (3DCERAM, now Solsi-Lux). The project, for which I designed a telescope presented in this work, has led to some exchanges and reflec-

tions about freeform optical design which were useful for the rest of the PhD.

We thank the people at Gaggione: David Gluchowski, Martin Berthel and Jean-Pierre Lauret. They manufactured (free of charge as part of their collaboration with academia on freeform optics) the freeform mirror sample which we measured and studied.

I want to thank Thales Alenia Space (TAS) and other entities of the Thales group (Angenieux and SESO) for taking part in the research work. We gratefully acknowledge their lending their facilities and the time of their engineers and technicians for the project. Hervé Benard (TAS) was instrumental in getting in touch with the right person at the right time in order for our research to progress. At TAS, we thank Fanny Keller for exchanges on optical design. We thank Nicolas Tetaz who agreed to participate in the PhD jury. We also thank, from the same company, Simon Henrot who contributed metrology data analysis included in our work and Jean-Philippe Sneed for attempting some metrology experiments. At SESO we thank Julien Fourez and Thomas Michel for having performed the LuphoScan measurement on the freeform sample as well as having lent the spherical BK7 caliber. We thank Francois Lepretre at Angenieux for lending the asphero-toroidal mirror.

We thank our colleagues Maxim Sakharov (Bauman MSTU), with whom we worked in order to investigate more advanced metaheuristics applied to optical design, and Charlie Vanaret (Technische Universität Berlin), researcher in another branch of applied mathematics and with whom we exchanged many ideas on subjects related to the present PhD work.

Finally we thank Jannick Rolland and Guillaume Druart, who took the time to review the present manuscript. They produced many insightful observations. I wish to thank the president of the jury, Yvan Sortais, who led the session masterfully.

Appendix A

Zernike Polynomials

Zernike polynomials are one of the most used polynomial bases. They are not the simplest to use however. It is useful to expatiate on the subject.

A.1 Definition

Zernike polynomials are defined using a set of conventions which we explain.

A.1.1 Coordinate System

The Zernike polynomials are defined over the unit disk. The explicit expression, as given by *e.g.* B&W [266], uses a polar coordinate system. We choose the coordinate system represented in Fig. A.1.

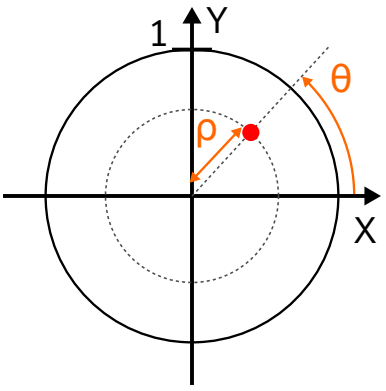


Figure A.1: Chosen coordinate system for the unit disk in cartesian and polar coordinates.

n	m	$Z_n^m(\rho, \theta)$
0	0	1
1	-1	$\rho \sin\theta$
1	1	$\rho \cos\theta$
2	-2	$\rho^2 \sin 2\theta$
2	0	$2\rho^2 - 1$
2	2	$\rho^2 \cos 2\theta$
3	-3	$\rho^3 \sin 3\theta$
3	-1	$(3\rho^3 - 2\rho) \sin\theta$
3	1	$(3\rho^3 - 2\rho) \cos\theta$
3	3	$\rho^3 \cos 3\theta$
...	...	...

Table A.1: First few Zernike polynomials using explicit expressions. We arbitrarily order them as in [77].

A.1.2 Double-indexing Scheme

Zernike polynomials can be indexed with a double-indexing scheme (n, m). n is called the *radial order* and m the *azimuthal order*. Following the recommandations of [77], we choose for our double-indexing:

- n: Starts at zero and iterates on positive integers: 0, 1, 2, 3, *etc.*
- m: Integer, for a given n, it can take the values such that $-n \leq m \leq n$ and $n + m$ is even. For example, for $n = 3$, $m = \{-3, -1, 1, 3\}$, for $n = 4$, $m = \{-4, -2, 0, 2, 4\}$, *etc.*

A.1.3 Explicit Expression

An explicit expression for Zernike polynomials was given in Eq. 2.24. These polynomials are *unit-normalized*. For each polynomial, the maximum absolute value over the unit disk is 1 [78]. Eq. 2.24 generates the explicit expressions of which the first few are written out in Tab. A.1. Such tables are found, albeit using different conventions, in [75, 77], in the OpticStudio user manual [68] and in many other works.

A.1.4 Single-indexing Scheme

A way of mapping the (n, m) double-indexing scheme onto a single index j is necessary. We choose to use Noll's indexing [267], which is the one used in OpticStudio's "Zernike Standard Sag". In Noll's indexing, the parity of the single index determines the choice of the azimuthal component $A_m(\theta)$ (between cosine and sine). In his paper [267], Noll did not give an explicit formula for his indexing scheme, but we can deduce an expression: Eq. A.1. Note that our indexing starts from 0, when Noll started it from 1.

$$j = \frac{n(n+1)}{2} + |m| - (\text{tests}(n,m) \bmod 2) \tag{A.1}$$

$$\text{tests}(n,m) = (n \bmod 2) + \left(\frac{n(n+1)}{2} \bmod 2\right) + \text{b}(m) \tag{A.1a}$$

$$\text{b}(m) = \begin{cases} 0 & m \leq 0 \\ 1 & m > 0 \end{cases} \tag{A.1b}$$

We provide useful equations when working with Zernike polynomials in Noll indexing [72]:

- Total number N_{terms} of polynomials for radial orders n from 0 up to n_{max}: Eq. A.2
- Radial order n corresponding to a given index j: Eq. A.3
- Azimuthal order m corresponding to a given index j: Alg. 1. The algorithm could also be expressed as a formula with cases, as in Eq. A.1.

$$N_{\text{terms}} = \frac{n_{\text{max}}(n_{\text{max}}+3)}{2} + 1 \tag{A.2}$$

$$n = \left\lceil \frac{-3 + \sqrt{9+8j}}{2} \right\rceil \tag{A.3}$$

A.1.5 Normalization

As noted by Andersen [78], the polynomials $Z_n^m(\rho,\theta)$ in Eq. 2.24 are *unit-normalized* in the sense of Eq. A.4.

$$\max_{\substack{0 \leq \rho \leq 1 \\ -\pi \leq \theta < \pi}} |Z_n^m(\rho,\theta)| = 1 \tag{A.4}$$

The polynomials used in OpticStudio are normalized to *unit standard deviation*. This is the normalization used by Noll [267]. We call these polynomials "Standard" after their name in OpticStudio. The standard Zernike polynomials $Z_{nm}^{\text{std}}(\rho,\theta)$ are defined by Eq. A.5.

$$Z_{nm}^{\text{std}}(\rho,\theta) = N_{nm} Z_n^m(\rho,\theta) \tag{A.5}$$

$$N_{nm} = \sqrt{(2-\delta_{m,0})(n+1)} \tag{A.5a}$$

Algorithm 1: Computing m from j in Noll indexing starting from 0.

```
input  : The index j.
output: The corresponding azimuthal order m.
npart ← n(n+1)/2;
mpart ← j - npart;
if mod(n,2) = 0 then
    if mod(npart,2) = 0 then
        if mod(mpart,2)=0 then
            m←-mpart;
        else
            m←mpart+1;
        end
    else
        if mod(mpart,2)=0 then
            m←mpart;
        else
            m←-mpart-1;
        end
    end
else
    if mod(npart,2) = 0 then
        if mod(mpart,2)=0 then
            m←-mpart-1;
        else
            m←mpart;
        end
    else
        if mod(mpart,2)=0 then
            m←mpart+1;
        else
            m←-mpart;
        end
    end
end
```

The meaning of *unit standard deviation*[1] is made explicit in Eq. A.6 (see eq.3.11 of Gray [75]).

$$\frac{1}{\pi}\int_{\theta=-\pi}^{\pi}\int_{\rho=0}^{1}(Z_{nm}^{\mathrm{std}}(\rho,\theta))^2\rho d\rho d\theta = 1 \tag{A.6}$$

A.1.6 Communicating Conventions

We know from experience that the different conventions used for Zernike polynomials can be the source of great confusion. Some domain-specific normalization effort has been going on [77] (ophtalmology) [268] (optical surface form tolerance)[2], but there is no consensus among optical design practitioners and manufacturers in practice. An exhaustive review of all the conventions used would be of limited interest but let us propose a list of items to pay attention to when working with Zernike polynomials:

- Coordinate system (see Fig. A.1)
  - What is the orientation of the axes X and Y?
  - What is the orientation for positive θ?
  - Is $\theta = 0$ on the X or Y axis?
- Polynomials definition
  - What is the double-indexing used? m can be positive only [78, 266, 267] or a signed integer [77].
  - How are cos and sin terms assigned? This can be dependent on the parity of the single-indexing [267] or on the sign of m [77].
- Single-indexing scheme
  - What single-indexing scheme is used? Noll [267], fringe, ANSI [77] and many more exist.
- Normalization
  - Are the polynomials *unit-normalized* (see Eq. 2.24) or *standard* (see Eq. A.5)? The normalization used in B&W [266] is yet another one.

We urge readers to communicate the conventions they use unambiguously. Cross-checking using point-cloud data is often necessary.

[1] Here, *standard deviation* is used in the sense of the square of the root-mean-square quantity. This is the mean of squares of a 2D function over its support.

[2] The conventions used in these two normative works differ.

A.1.7 Polynomials Orthogonality

Zernike standard polynomials are orthonormal over the unit-disk (eq. III.5 of Gray [75] and eq. 7 of [72]), Eq. A.7.

$$\frac{1}{\pi}\int_{\theta=-\pi}^{\pi}\int_{\rho=0}^{1} Z_{nm}^{\text{std}}(\rho,\theta)\cdot Z_{n'm'}^{\text{std}}(\rho,\theta)\rho d\rho d\theta = \delta_{n,n'}\delta_{m,m'} \tag{A.7}$$

A.1.8 Using the Normalization Radius, Extrapolation and Aperture Shape

In optical design, it is often useful to deviate slightly from canonical Zernike polynomials.

A.1.8.1 Normalization Radius

The outer ring of the unit disk on Zernike polynomials of a higher $n+m$ order is rarely useful in optical design (see the terms $j = 43, 44$ in Fig. 2.7). These high amplitude variations on the edge of the surface are likely to make the system behave in a chaotic way (especially computations involving marginal rays). This is why optical designers will often crop this part by using a normalization radius greater than the clear aperture of the optical surfaces. This modified table of polynomials can be seen in Fig. A.2. We plotted a clear aperture that is only 80 % of the normalization radius. As can be seen, with this designer's trick, some terms that represented mainly high spatial frequencies on the outer edge of the surface are now almost flat (*eg* 19;5;5) and the troublesome outer edge of some other terms is cropped out (*eg* 23;6;2).

Another tip is to use Zernike polynomials in increasing order of $n + m$, to better match the order in which optical aberrations are likely to appear. This idea is used in the so-called *Fringe* ordering.

A.1.8.2 Aperture Shape and Extrapolation

The Zernike polynomials can be adapted to elliptical, annular or arbitrary aperture shape while retaining orthogonality with a Gram-Schmidt orthogonalization [72]. Additionally, the polynomials can also be extrapolated outside of the unit disk to fit rectangular aperture shapes for example, although extrapolation will likely generate important departures in altitude and slope.

A.2 Practical Implementation

Having an implementation independent from off-the-shelf optical design tool is useful for cross-checking results and working with external tools on the shape of freeform surfaces. A table with the first Zernike polynomials is given in Fig. 2.7.

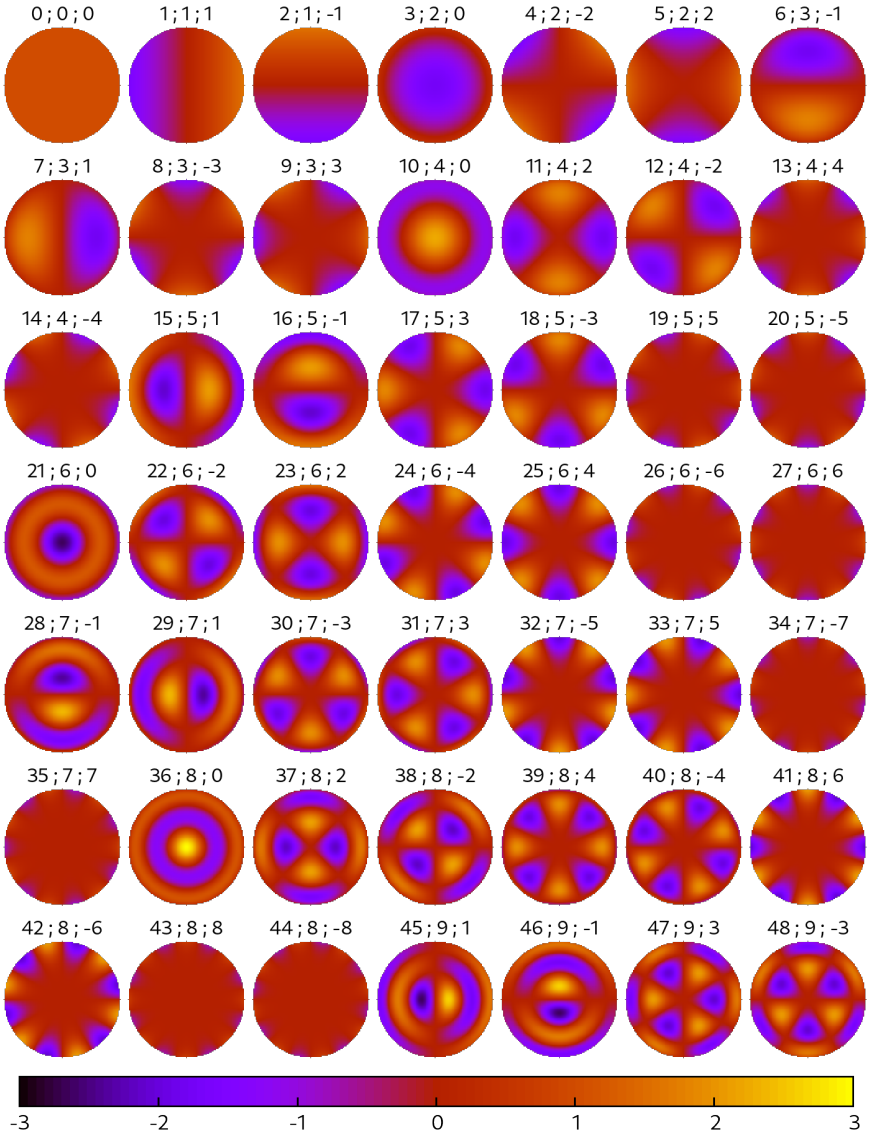


Figure A.2: Table of the first few Zernike polynomials in OpticStudio "Zernike Standard Sag" indexing and normalization. The clear aperture is only $80\,\%$ of the radius of normalization. The color scale is common to all the polynomials. We give numbers j, n, m as in Fig. 2.7.

We elaborate on the implementation of Zernike polynomials and cartesian first derivatives using the work of Andersen [78].

A.2.1 Works on Recurrence Relations

As already mentioned, the explicit expression given in Eq. 2.24 cannot be used for a computer implementation when high orders must be computed due to a lack of numerical accuracy. An additional goal for implementations is to compute the first spatial derivatives of the Zernike polynomials, as they are required for raytracing. Many recurrence relations have been derived to circumvent the problem and compute higher order Zernike polynomials while preserving numerical accuracy and execution speed as much as possible [90, 269–271]. A noteworthy alternative to recurrence relations is mentioned in [72]: the computation of higher orders can make use of the Fourier transform to improve the numerical accuracy.

Andersen [78] has provided a recurrence relation that is both accurate and efficient for the computation of Zernike polynomials and their first derivatives in a cartesian coordinate system. He provided pseudo-code that makes the implementation straightforward. His work was implemented in OpticStudio 20.1 [100].

A.2.2 Implementation

We wrote our own implementation of Andersen's work [78] in Common Lisp and validated it against the established implementation in OpticStudio. We give some formulae that allow generating Zernike polynomials using the same conventions as in OpticStudio's "Zernike Standard Sag".

A.2.2.1 Conventions Conversion

Given the coordinate system in Fig. A.1, Andersen's pseudocode produces Zernike polynomials and cartesian derivatives that are *unit-normalized* Eq. A.4, and use a single-indexing that can also be found in [72]. One can work with the single-index j_{And} using Eqs. A.8 and A.9. Eqs. A.2 and A.3 still apply. By applying the indexing conversion as well as the normalization term Eq. A.5, one can obtain Zernike polynomials and derivatives exactly as computed in OpticStudio.

$$j_{And} = \frac{n(n+2)+m}{2} \tag{A.8}$$

$$m = 2j_{And} - n(n+2) \tag{A.9}$$

A.2.2.2 Validation methodology

We drew 231[3] random coefficients using a uniform distribution in the interval $[-1, 1]$. 100 random points in the unit disk were also drawn. We independently evaluated the altitude, X derivative and Y derivative of the generated "Zernike Standard Sag" surface at each random point on the unit disk using either Zemax 13 SP1 EE Version April 4th, 2013[4] or our own implementation. We validated our implementation using this method. Additionally, a visual check on the produced Zernike polynomials maps in Fig. 2.7 is also useful.

[3] 231 coefficients is the maximum number of terms supported the "Zernike Standard Sag" surface type in OpticStudio for the version we used.

[4] We did not use the latest OpticStudio due to limited licence availability at Sophia Engineering, this older version is sufficient for validation as long as the last decimals of accuracy are not considered, as assessed by [78].

Appendix B

Questions that come up concerning aperture stops in raytracing programs

B.1 Altitude across the aperture stop

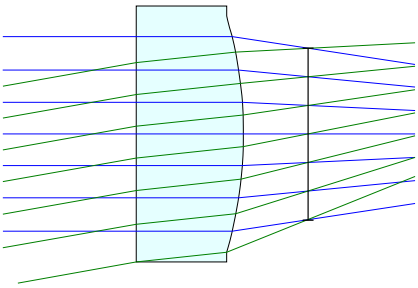
When the aperture stop of an optical system is an iris diaphragm, such as is often the case in photographic objectives, then there is no question as to where the rays should cross on the surface: they cross on the aperture plane (Fig. B.1a).

One may however be left wondering what to do in the case where no mechanical diaphragm is present and the light cones in the system are limited by an optical surface. What shape should the aperture stop surface have in this case? Couldn't we use a planar aperture stop at the apex of our optical surface (Fig. B.1b)? Or with its outer limits aligned with those of the optical surface (Fig. B.1c)? And what to do with more pathological freeform surfaces (Fig. B.1d)?

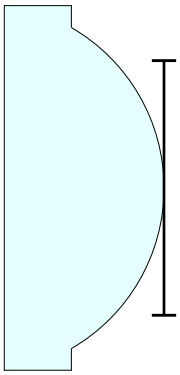
Let us explore what would happen if we were to perform raytracing on a planar aperture stop positioned on top of an optical surface. Geometrically, the stop would in many cases cross the optical surface, with some portion of the surface positioned behind the stop and the rest in front of it. This would happen in every case for a S-shaped lens (Fig. B.1d) but let us look at a planar aperture stop crossing a convex lens for the sake of simplicity. In sequential raytracing, we have two options: the stop can be placed in the raytracing surfaces sequence either before or after the optical surface we set it on, independently of the geometry. When the stop is set logically before the surface (Fig. B.2a), the ray bundles for the different fields cross at the aperture stop position before having been affected by the refraction. When the stop is set after the surface (Fig. B.2b), the ray bundles cross at the aperture stop after the refraction has been applied. When tracing single rays, we then have four possible cases (Figs. B.2c to B.2f)

Figure B.1: Different intuitions for setting the position of the aperture stop when a planar shape is assumed.

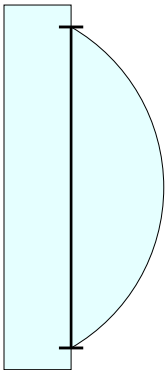
(a) An aperture stop on a mechanical iris diaphragm after a lens. The ray bundles corresponding to each field cross at the aperture stop sampling points along the surface plane.



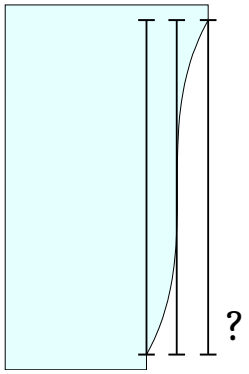
(b) A planar aperture stop positioned at the apex of a convex lens.



(c) A planar aperture stop with its border aligned to the outer limit of a convex lens.



(d) A planar aperture stop on a somewhat pathological, but realistic, S-shaped surface.



depending on whether the stop is before or after the surface in the raytracing chain and geometrically along the ray. In every case, the ray-aiming is made to work unambiguously: ray bundles from different fields cross at the points defined by the aperture stop sampling on the planar stop surface, even though some rays are virtual (Algs. 2 and 3). However, we see that the rays that exit the optical surface after the refraction are necessarily different depending on whether we choose to place the aperture stop before or after the surface in the raytracing sequence. This makes little sense if we think about our optical system as being real and not simulated: we have an aperture stop that is a physical object with a fixed geometrical position and our simulation result is influenced by whether we choose to locate our aperture stop in the object space of our optical surface or in its image space.

Algorithm 2: Direct raytracing sequence when the stop is in the object space of the optical surface.

1: Intersect with the stop.
2: Intersect with the surface.
3: Refract by the surface.

Algorithm 3: Direct raytracing sequence when the stop is in the image space of the optical surface.

1: Intersect with the surface.
2: Refract by the surface.
3: Intersect with the stop.

The solution that was adopted by raytracing programs is to make the aperture stop follow the altitude of the optical surfaces when they bear the stop (Figs. B.3a and B.3b). This is, as far as we know, the only solution that makes sense. The optical surface itself is the only physical object that is the intersection between the object space and image space of itself. It is the only place where the choice of raytracing space (object or image space of the optical surface) for the aperture stop does not affect the resulting rays traced.[1]

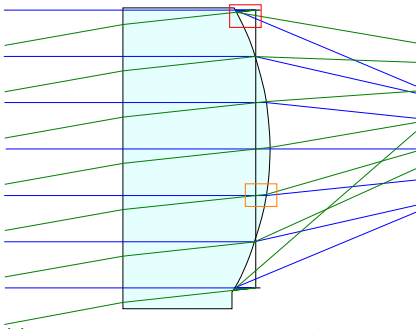
B.2 Real optical system with aperture stop across two raytracing spaces

We can imagine an optical system with an aperture stop position that would make raytracing quite hard for sequential raytracing programs. An example of such a system is given in Fig. B.4. In this system, the rays follow a different raytracing sequence depending on their position in the aperture stop sampling. We give the two different, complete, direct raytracing sequences in Algs. 4 and 5.

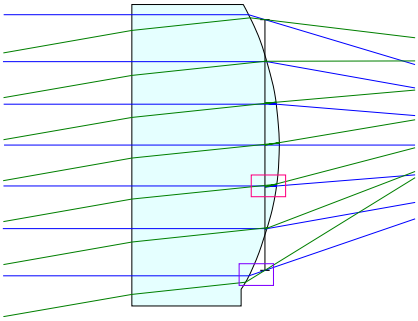
[1] We can think of esoteric systems where a physical iris is indeed across two raytracing spaces (Appendix B.2).

Figure B.2: Raytracing using a planar aperture stop crossing a convex optical surface in two cases: stop placed before or after the surface in the raytracing chain.

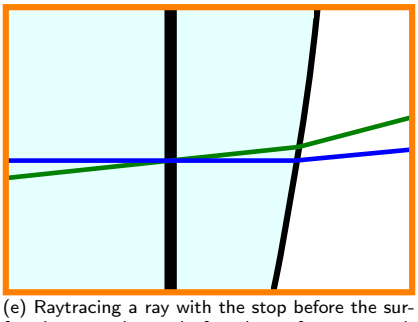
(a) Raytracing with stop placed before the surface in the raytracing.



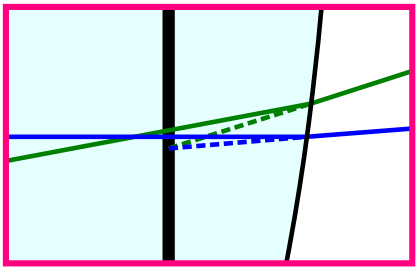
(b) Raytracing with stop placed after the surface in the raytracing.



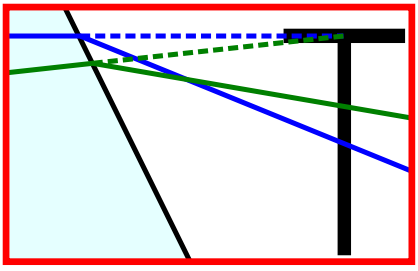
(c) Raytracing a ray with the stop before the surface in raytracing and before the surface geometrically.



(d) Raytracing a ray with the stop after the surface in raytracing and before the surface geometrically.



(e) Raytracing a ray with the stop before the surface in raytracing and after the surface geometrically.



(f) Raytracing a ray with the stop after the surface in raytracing and after the surface geometrically.

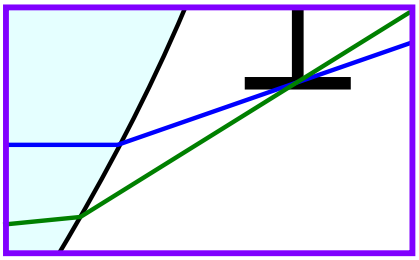


Figure B.3: An aperture stop positioned on the optical surface on two lenses.

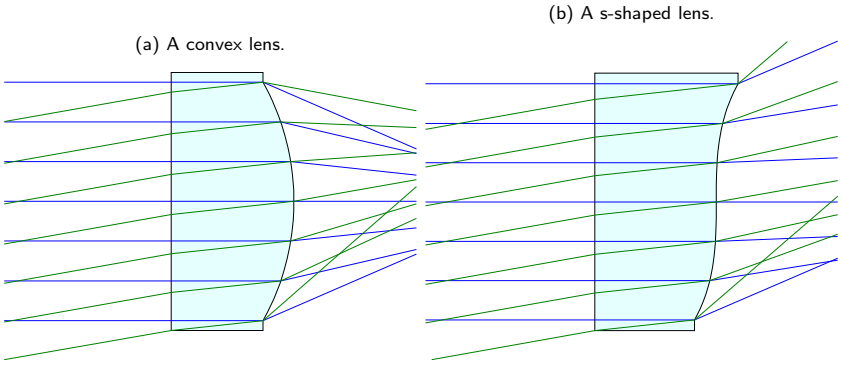
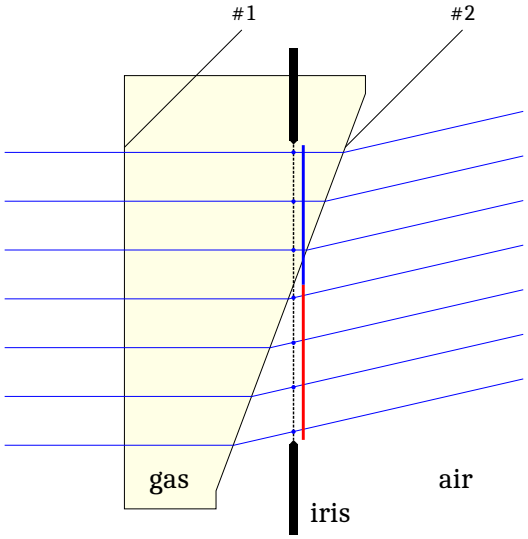


Figure B.4: A hypothetical gas lens with an iris diaphragm. The iris is the aperture stop of the optical system. The rays belonging to the blue zone of the aperture stop follow a different raytracing sequence than those belonging to the red zone. #1 and #2 denote the two air/gas interfaces.



Algorithm 4: Direct ray-tracing sequence for the rays in the blue (gas) zone of the aperture stop in the hypothetical gas lens system.

1: Intersect with the surface #1.
2: Refract by the surface #1.
3: **Intersect with the aperture stop surface.**
4: Intersect with the surface #2.
5: Refract by the surface #2.

Algorithm 5: Direct ray-tracing sequence for the rays in the red (air) zone of the aperture stop.

1: Intersect with the surface #1.
2: Refract by the surface #1.
3: Intersect with the surface #2.
4: Refract by the surface #2.
5: **Intersect with the aperture stop surface.**

Bibliography

[1] Jannick P. Rolland et al. "Freeform optics for imaging". In: *Optica* 8.2 (Jan. 2021), p. 161. DOI: 10.1364/optica.413762.

[2] William T. Plummer. "Free-form optical components in some early commercial products". In: *Tribute to Warren Smith: A Legacy in Lens Design and Optical Engineering*. Ed. by Robert E. Fischer. Vol. 5865. International Society for Optics and Photonics. SPIE, 2005, pp. 67–73. DOI: 10.1117/12.623875.

[3] Bernard Maitenaz. "Four steps that led to Varilux". In: *Am J Optom Arch Am Acad Optom.* 7 (43 July 1966). DOI: 10.1097/00006324-196607000-00004.

[4] Luis W Alvarez. "Two-element variable-power spherical lens". US3305294A. US Patent. Dec. 1964.

[5] Ozan Cakmakci and Jannick Rolland. "Design and fabrication of a dual-element off-axis near-eye optical magnifier". In: *Optics Letters* 32.11 (Apr. 2007), p. 1363. DOI: 10.1364/ol.32.001363.

[6] Daniel K. Nikolov et al. "Metaform optics: Bridging nanophotonics and freeform optics". In: *Science Advances* 7.18 (Apr. 2021), eabe5112. DOI: 10.1126/sciadv.abe5112.

[7] Cheng Dewen, Hong Hua, and Yongtian Wang. "Optical see-through free-form head-mounted display". 13/318,864. US Patent. Apr. 2012.

[8] Peter Ott. "Optic design of head-up displays with freeform surfaces specified by NURBS". In: *Optical Design and Engineering III*. Ed. by Laurent Mazuray et al. Vol. 7100. International Society for Optics and Photonics. SPIE, 2008, pp. 339–350. DOI: 10.1117/12.797761.

[9] Juan C. Miñano, Pablo Benítez, and Asunción Santamaría. "Free-form optics for illumination". In: *Optical Review* 16 (2 Mar. 2009), pp. 99–102. DOI: 10.1007/s10043-009-0017-4.

[10] A. Zuccaro Marchi, L. Maresi, and M. Taccola. "Technologies and designs for small optical missions". In: *International Conference on Space Optics — ICSO 2018*. Ed. by Zoran Sodnik, Nikos Karafolas, and Bruno Cugny. Vol. 11180. International Society for Optics and Photonics. SPIE, 2019, pp. 750–759. DOI: 10.1117/12.2535990.

[11] H. Philip Stahl et al. "Survey of interferometric techniques used to test JWST optical components". In: *Interferometry XV: Techniques and Analysis*. Ed. by Catherine E. Towers, Joanna Schmit, and Katherine Creath. Vol. 7790. International Society for Optics and Photonics. SPIE, 2010, pp. 15–27. DOI: 10.1117/12.862234.

[12] Kevin P. Thompson et al. "Using nodal aberration theory to understand the aberrations of multiple unobscured three mirror anastigmatic (TMA) telescopes". In: *Optical System Alignment, Tolerancing, and Verification III*. Ed. by José Sasián and Richard N. Youngworth. Vol. 7433. International Society for Optics and Photonics. SPIE, 2009, pp. 104–111. DOI: 10.1117/12.826749.

[13] T. Peschel et al. "Design of an imaging spectrometer for earth observation using freeform mirrors". In: *International Conference on Space Optics — ICSO 2016*. Ed. by Bruno Cugny, Nikos Karafolas, and Zoran Sodnik. Vol. 10562. International Society for Optics and Photonics. SPIE, 2017, pp. 928–936. DOI: 10.1117/12.2296156.

[14] ESA. *ISS Utilization: MUSES-DESIS (Multi-User System for Earth Sensing) with DESIS instrument*. [Online; accessed 25-Nov-2020]. URL: https://directory.eoportal.org/web/eoportal/satellite-missions/content/-/article/iss-muses.

[15] David Nijkerk et al. "The TROPOMI Telescope". In: *International Conference on Space Optics — ICSO 2012*. Ed. by Bruno Cugny, Errico Armandillo, and Nikos Karafolas. Vol. 10564. International Society for Optics and Photonics. SPIE, 2017, pp. 272–278. DOI: 10.1117/12.2309035.

[16] Roland Geyl et al. “Freeform optics design and manufacturing for MicroCarb”. In: *Optical Fabrication, Testing, and Metrology VI*. Ed. by Sven Schröder and Roland Geyl. SPIE, June 2018. DOI: 10.1117/12.2313243.
[17] GW Ritchey. “The Modern Photographic Telescope and the New Astronomical Photography-Part III-The Ritchey-Chrétien Aplanatic Reflector”. In: *Journal of the Royal Astronomical Society of Canada* 22 (1928), p. 303. URL: http://adsabs.harvard.edu/full/1928JRASC..22..303R.
[18] Museo Galileo. *Galileo's telescope*. [Online; accessed 11-Nov-2020]. URL: https://catalogue.museogalileo.it/object/GalileosTelescope_n01.html.
[19] Paul Feyerabend. *Against Method: Outline of an Anarchistic Theory of Knowledge*. 1975. ISBN: 9780391003811.
[20] Andrea Del Centina. “On Kepler's system of conics in Astronomiae pars optica”. In: *Archive for History of Exact Sciences* 70 (6 Nov. 2016), pp. 567–589. DOI: 10.1007/s00407-016-0175-2.
[21] André Baranne and Françoise Launay. “Cassegrain: a famous unknown of instrumental astronomy”. In: *Journal of Optics* 28.4 (Aug. 1997), pp. 158–172. DOI: 10.1088/0150-536x/28/4/004.
[22] Jean Texereau. *La construction du téléscope d'amateur*. 2nd ed. Vuibert, 2017. ISBN: 978-2711753437.
[23] J. G. Baker. “On Improving the Effectiveness of Large Telescopes”. In: *IEEE Transactions on Aerospace and Electronic Systems* AES-5.2 (1969), pp. 261–272. DOI: 10.1109/TAES.1969.309914.
[24] Dietrich Korsch. “Closed Form Solution for Three-Mirror Telescopes, Corrected for Spherical Aberration, Coma, Astigmatism, and Field Curvature”. In: *Appl. Opt.* 11.12 (Dec. 1972), pp. 2986–2987. DOI: 10.1364/AO.11.002986.
[25] Dietrich Korsch. “Anastigmatic three-mirror telescope”. In: *Appl. Opt.* 16.8 (Aug. 1977), pp. 2074–2077. DOI: 10.1364/AO.16.002074.
[26] Wikipedia contributors. *40-foot telescope — Wikipedia, The Free Encyclopedia*. [Online; accessed 14-November-2020]. 2020. URL: https://en.wikipedia.org/w/index.php?title=40-foot_telescope&oldid=967439760.
[27] Anton Kutter. *The Schiefspiegler (Oblique Telescope)*. Sky Pub., 1958.
[28] Kyle Fuerschbach. “Freeform, phi-polynomial optical surfaces: optical design, fabrication and assembly”. PhD thesis. University of Rochester, 2014. URL: http://hdl.handle.net/1802/28531.
[29] Lacy G. Cook. “Three mirror anastigmatic optical system”. US4265510A. US Patent. May 1979.
[30] Jose Sasian. *Introduction to aberrations in optical imaging systems*. Cambridge University Press, 2013.
[31] Jannick P. Rolland et al. “Roland V. Shack's discovery of nodal aberration theory, the expansion into the aberrations of freeform optics, and impact in optical design”. In: *Roland V. Shack Memorial Session: A Celebration of One of the Great Teachers of Optical Aberration Theory*. Ed. by John P. Lehan. SPIE, Aug. 2020. DOI: 10.1117/12.2570794.
[32] Harold Horace Hopkins. *Wave theory of aberrations*. Clarendon Press, 1950.
[33] Kevin P. Thompson. “Multinodal fifth-order optical aberrations of optical systems without rotational symmetry: spherical aberration”. In: *J. Opt. Soc. Am. A* 26.5 (May 2009), pp. 1090–1100. DOI: 10.1364/JOSAA.26.001090.
[34] Richard Alfred Buchroeder. “Tilted component optical systems”. 1976.
[35] Roland V. Shack and Kevin Thompson. “Influence Of Alignment Errors Of A Telescope System On Its Aberration Field”. In: *Optical Alignment I*. Ed. by Richard N. Shagam and William C. Sweatt. Vol. 0251. International Society for Optics and Photonics. SPIE, 1980, pp. 146–153. DOI: 10.1117/12.959464.
[36] Kevin P. Thompson. “Multinodal fifth-order optical aberrations of optical systems without rotational symmetry: the comatic aberrations”. In: *J. Opt. Soc. Am. A* 27.6 (June 2010), pp. 1490–1504. DOI: 10.1364/JOSAA.27.001490.
[37] Kevin P. Thompson. “Multinodal fifth-order optical aberrations of optical systems without rotational symmetry: the astigmatic aberrations”. In: *J. Opt. Soc. Am. A* 28.5 (May 2011), pp. 821–836. DOI: 10.1364/JOSAA.28.000821.
[38] Kevin P. Thompson et al. “Recent Discoveries from Nodal Aberration Theory”. In: *International Optical Design Conference and Optical Fabrication and Testing*. Optical Society of America, 2010, IThB2. DOI: 10.1364/IODC.2010.IThB2.
[39] Kyle Fuerschbach, Jannick P. Rolland, and Kevin P. Thompson. “Theory of aberration fields for general optical systems with freeform surfaces”. In: *Optics Express* 22.22 (Oct. 2014), p. 26585. DOI: 10.1364/oe.22.026585.

[40] Bauer Aaron, Schiesser Eric, and Rolland Jannick. "Starting geometry creation and design method for freeform optics". In: *Nature Communications* 9.1756 (May 2018). DOI: 10.1038/s41467-018-04186-9.

[41] F.Z. Fang et al. "Manufacturing and measurement of freeform optics". In: *CIRP Annals* 62.2 (2013), pp. 823–846. ISSN: 0007-8506. DOI: 10.1016/j.cirp.2013.05.003.

[42] Fengzhou Fang, Nan Zhang, and Xiaodong Zhang. "Precision injection molding of freeform optics". In: *Advanced Optical Technologies* 5.4 (2016), pp. 303–324.

[43] Zexiao Li et al. "Machining approach of freeform optics on infrared materials via ultra-precision turning". In: *Opt. Express* 25.3 (Feb. 2017), pp. 2051–2062. DOI: 10.1364/OE.25.002051.

[44] Xiaodong Zhang et al. "Machining of optical freeform prisms by rotating tools turning". In: *CIRP Annals* 61.1 (2012), pp. 519–522. ISSN: 0007-8506. DOI: 10.1016/j.cirp.2012.03.009.

[45] S. Grabarnik et al. "Compact multispectral and hyperspectral imagers based on a wide field of view TMA". In: *International Conference on Space Optics — ICSO 2010*. Ed. by Errico Armandillo, Bruno Cugny, and Nikos Karafolas. Vol. 10565. International Society for Optics and Photonics. SPIE, 2017, pp. 27–33. DOI: 10.1117/12.2309101.

[46] Yazid Tohme, Ralph Murray, and E Allaire. "Principles and applications of the slow slide servo". In: *Moore Nanotechnology Systems White Paper* (2005).

[47] Zexiao Li et al. "Highly efficient machining of non-circular freeform optics using fast tool servo assisted ultra-precision turning". In: *Opt. Express* 25.21 (Oct. 2017), pp. 25243–25256. DOI: 10.1364/OE.25.025243.

[48] Shoichi Yamazaki et al. "Thin wide-field-of-view HMD with free-form-surface prism and applications". In: *Stereoscopic Displays and Virtual Reality Systems VI*. Ed. by Mark T. Bolas et al. Vol. 3639. International Society for Optics and Photonics. SPIE, 1999, pp. 453–462. DOI: 10.1117/12.349411.

[49] Melanie Roulet et al. "Off-axis parabolas super polished under stress: the case of the Roman Space Telescope coronagraphic instrument mirrors". In: *Opt. Express* 28.21 (Oct. 2020), pp. 30555–30569. DOI: 10.1364/OE.400284.

[50] Donald Golini et al. "Magnetorheological finishing (MRF) in commercial precision optics manufacturing". In: *Optical Manufacturing and Testing III*. Ed. by H. Philip Stahl. Vol. 3782. International Society for Optics and Photonics. SPIE, 1999, pp. 80–91. DOI: 10.1117/12.369174.

[51] S. R. Wilson and J. R. McNeil. "Neutral Ion Beam Figuring Of Large Optical Surfaces". In: *Current Developments in Optical Engineering II*. Ed. by Robert E. Fischer and Warren J. Smith. Vol. 0818. International Society for Optics and Photonics. SPIE, 1987, pp. 320–324. DOI: 10.1117/12.978903.

[52] Martin Weiser. "Ion beam figuring for lithography optics". In: *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms* 267.8 (2009). Proceedings of the 16th International Conference on Ion Beam Modification of Materials, pp. 1390–1393. ISSN: 0168-583X. DOI: 10.1016/j.nimb.2009.01.051.

[53] Harrison Herzog et al. "Optical fabrication of lightweighted 3D printed mirrors". In: *Optomechanical Engineering 2015*. Ed. by Alson E. Hatheway. Vol. 9573. International Society for Optics and Photonics. SPIE, 2015, pp. 53–67. DOI: 10.1117/12.2188197.

[54] Nina Vaidya and Olav Solgaard. "3D printed optics with nanometer scale surface roughness". In: *Microsystems & Nanoengineering* 4 (1 2018), p. 18. DOI: 10.1038/s41378-018-0015-4.

[55] Enrico Hilpert et al. "Design, additive manufacturing, processing, and characterization of metal mirror made of aluminum silicon alloy for space applications". In: *Optical Engineering* 58.9 (2019), pp. 1–9. DOI: 10.1117/1.OE.58.9.092613.

[56] Ksenia Weber et al. "Distortion-free multi-element Hypergon wide-angle micro-objective obtained by femtosecond 3D printing". In: *Opt. Lett.* 45.10 (May 2020), pp. 2784–2787. DOI: 10.1364/OL.392253.

[57] Nisrine Louh. "Optical mirrors design using topology optimization for additive manufacturing (Conference Presentation)". In: *Material Technologies and Applications to Optics, Structures, Components, and Sub-Systems IV*. Ed. by Matthias Kroedel and Bill A. Goodman. Vol. 11101. International Society for Optics and Photonics. SPIE, 2019. DOI: 10.1117/12.2529515.

[58] Tobias Schmid, Kevin Thompson, and Jannick Rolland. "Alignment of two mirror astronomical telescopes (the astigmatic component)". In: *Modeling, Systems Engineering, and Project Management for Astronomy III*. Ed. by George Z. Angeli and Martin J. Cullum. Vol. 7017. International Society for Optics and Photonics. SPIE, 2008, pp. 111–122. DOI: 10.1117/12.791263.

[59] Tobias Schmid, Kevin P. Thompson, and Jannick P. Rolland. "A unique astigmatic nodal property in misaligned Ritchey-Chrétien telescopes with misalignment coma removed". In: *Opt. Express* 18.5 (Mar. 2010), pp. 5282–5288. DOI: 10.1364/OE.18.005282.

[60] Kevin Thompson, Tobias Schmid, and Jannick Rolland. "Alignment induced aberration fields of next generation telescopes". In: *Optical Design and Testing III*. Ed. by Yongtian Wang et al. Vol. 6834. International Society for Optics and Photonics. SPIE, 2007, pp. 100–108. DOI: 10.1117/12.772010.

[61] Karen M. Hampson et al. "High precision automated alignment procedure for two-mirror telescopes". In: *Appl. Opt.* 58.27 (Sept. 2019), pp. 7388–7391. DOI: 10.1364/AO.58.007388.

[62] D. Scott Acton et al. "Wavefront sensing and controls for the James Webb Space Telescope". In: *Space Telescopes and Instrumentation 2012: Optical, Infrared, and Millimeter Wave*. Ed. by Mark C. Clampin et al. Vol. 8442. International Society for Optics and Photonics. SPIE, 2012, pp. 877–887. DOI: 10.1117/12.925015.

[63] H. Gross et al. "Overview of surface representations for freeform surfaces". In: *Optical Systems Design 2015: Optical Design and Engineering VI*. Ed. by Laurent Mazuray, Rolf Wartmann, and Andrew P. Wood. Vol. 9626. International Society for Optics and Photonics. SPIE, 2015, pp. 195–208. DOI: 10.1117/12.2191255.

[64] Jingfei Ye et al. "Review of optical freeform surface representation technique and its application". In: *Optical Engineering* 56.11 (2017), pp. 1–15. DOI: 10.1117/1.OE.56.11.110901.

[65] Ralf Steinkopf et al. "Data handling and representation of freeform surfaces". In: *Optical Fabrication, Testing, and Metrology IV*. Ed. by Angela Duparré and Roland Geyl. Vol. 8169. International Society for Optics and Photonics. SPIE, 2011, pp. 191–199. DOI: 10.1117/12.896848.

[66] Anika Brömel, Uwe Lippmann, and Herbert Gross. "Freeform surface descriptions. Part I: Mathematical representations". In: *Advanced Optical Technologies* 6.5 (2017), pp. 327–336. DOI: 10.1515/aot-2017-0030.

[67] Anika Brömel. "Development and evaluation of freeform surface descriptions". en. Dissertation, Friedrich-Schiller-Universität Jena, 2018. PhD thesis. Jena, 2018. DOI: 10.22032/dbt.38072.

[68] Zemax LLC, ed. *OpticStudio User Manual*. Version Version 20.1. Jan. 2020.

[69] Nick Takaki, Aaron Bauer, and Jannick P. Rolland. "Degeneracy in freeform surfaces described with orthogonal polynomials". In: *Applied Optics* 57.35 (Dec. 2018), pp. 10348–10354. DOI: 10.1364/AO.57.010348.

[70] Wikipedia contributors. *Smoothstep — Wikipedia, The Free Encyclopedia*. [Online; accessed 29-April-2020]. 2020. URL: https://en.wikipedia.org/w/index.php?title=Smoothstep&oldid=949917043.

[71] Wikipedia contributors. *Sigmoid function — Wikipedia, The Free Encyclopedia*. [Online; accessed 29-April-2020]. 2020. URL: https://en.wikipedia.org/w/index.php?title=Sigmoid_function&oldid=953781931.

[72] Vasudevan Lakshminarayanan and Andre Fleck. "Zernike polynomials: a guide". In: *Journal of Modern Optics* 58.7 (2011), pp. 545–561. DOI: 10.1080/09500340.2011.554896.

[73] Wikipedia contributors. *Inner product space — Wikipedia, The Free Encyclopedia*. [Online; accessed 12-June-2020]. 2020. URL: https://en.wikipedia.org/w/index.php?title=Inner_product_space&oldid=962020729.

[74] Anika Broemel et al. "Freeform surface descriptions. Part II: Application benchmark". In: *Advanced Optical Technologies* 6.5 (2017), pp. 337–347. DOI: 10.1515/aot-2017-0031.

[75] Robert W. Gray. "Investigation of the Field Dependence of the Aberration Functions of Rotationally Nonsymmetric Optical Imaging Systems". PhD thesis. University of Rochester, 2015. URL: http://hdl.handle.net/1802/30415.

[76] Ilhan Kaya, Kevin P. Thompson, and Jannick P. Rolland. "Edge clustered fitting grids for ϕ-polynomial characterization of freeform optical surfaces". In: *Opt. Express* 19.27 (Dec. 2011), pp. 26962–26974. DOI: 10.1364/OE.19.026962.

[77] Larry N. Thibos et al. "Standards for Reporting the Optical Aberrations of Eyes". In: *Vision Science and its Applications*. Optical Society of America, 2000, SuC1. DOI: 10.1364/VSIA.2000.SuC1.

[78] Torben B. Andersen. "Efficient and robust recurrence relations for the Zernike circle polynomials and their derivative s in Cartesian coordinates". In: *Opt. Express* 26.15 (July 2018), pp. 18878–18896. DOI: 10.1364/OE.26.018878.

[79] Stef Graillat, Vincent Lefèvre, and Jean-Michel Muller. "On the maximum relative error when computing integer powers by iterated multiplications in floating-point arith-

metic". In: *Numerical Algorithms* 70 (Feb. 2015), pp. 653–667. DOI: 10.1007/s11075-015-9967-8.

[80] Yongjun Xie et al. "Optical design and fabrication of an all-aluminum unobscured two-mirror freeform imaging telescope". In: *Appl. Opt.* 59.3 (Jan. 2020), pp. 833–840. DOI: 10.1364/AO.379324.

[81] Zhengxiang Shen et al. "Customized design and efficient fabrication of two freeform aluminum mirrors by single point diamond turning technique". In: *Appl. Opt.* 58.9 (Mar. 2019), pp. 2269–2276. DOI: 10.1364/AO.58.002269.

[82] G.W. Forbes. "Characterizing the shape of freeform optics". In: *Opt. Express* 20.3 (Jan. 2012), pp. 2483–2499. DOI: 10.1364/OE.20.002483.

[83] G.W. Forbes. "Shape specification for axially symmetric optical surfaces". In: *Opt. Express* 15.8 (Apr. 2007), pp. 5218–5226. DOI: 10.1364/OE.15.005218.

[84] G.W. Forbes. "Robust, efficient computational methods for axially symmetric optical aspheres". In: *Opt. Express* 18.19 (Sept. 2010), pp. 19700–19712. DOI: 10.1364/OE.18.019700.

[85] Eric W. Weisstein. *Legendre Polynomial*. From MathWorld–A Wolfram Web Resource [Online; accessed 03-Jun-2020]. 2020. URL: https://mathworld.wolfram.com/LegendrePolynomial.html.

[86] Eric W. Weisstein. *Chebyshev Polynomial of the First Kind*. From MathWorld–A Wolfram Web Resource [Online; accessed 05-Jun-2020]. 2020. URL: https://mathworld.wolfram.com/ChebyshevPolynomialoftheFirstKind.html.

[87] Eric W. Weisstein. *Chebyshev Polynomial of the Second Kind*. From MathWorld–A Wolfram Web Resource [Online; accessed 05-Jun-2020]. 2020. URL: https://mathworld.wolfram.com/ChebyshevPolynomialoftheSecondKind.html.

[88] M. Gray, M. Ferrari, and S. Vives. "A new basis of polynomials for off-axis highly aspheric optical surfaces". In: *Optical Complex Systems: OCS11*. Ed. by Gérard Berginc. Vol. 8172. International Society for Optics and Photonics. SPIE, 2011, pp. 284–291. DOI: 10.1117/12.896629.

[89] S. Pascal et al. "New modelling of freeform surfaces for optical design of astronomical instruments". In: *Modern Technologies in Space- and Ground-based Telescopes and Instrumentation II*. Ed. by Ramón Navarro, Colin R. Cunningham, and Eric Prieto. Vol. 8450. International Society for Optics and Photonics. SPIE, 2012, pp. 1650–1659. DOI: 10.1117/12.925299.

[90] G. W. Forbes. "Robust and fast computation for the polynomials of optics". In: *Optics Express* 18.13 (June 2010), pp. 13851–13862. DOI: 10.1364/OE.18.013851.

[91] *NIST Digital Library of Mathematical Functions*. http://dlmf.nist.gov/, Release 1.0.26 of 2020-03-15. F. W. J. Olver, A. B. Olde Daalhuis, D. W. Lozier, B. I. Schneider, R. F. Boisvert, C. W. Clark, B. R. Miller, B. V. Saunders, H. S. Cohl, and M. A. McClain, eds. URL: http://dlmf.nist.gov/.

[92] William H Press et al. *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press, 2007.

[93] Wikipedia contributors. *Kahan summation algorithm — Wikipedia, The Free Encyclopedia*. [Online; accessed 11-June-2020]. 2020. URL: https://en.wikipedia.org/w/index.php?title=Kahan_summation_algorithm&oldid=953520975.

[94] Wikipedia contributors. *Estrin's scheme — Wikipedia, The Free Encyclopedia*. [Online; accessed 11-June-2020]. 2018. URL: https://en.wikipedia.org/w/index.php?title=Estrin%27s_scheme&oldid=869528955.

[95] Eric W. Weisstein. *Jacobi Polynomial*. From MathWorld–A Wolfram Web Resource [Online; accessed 12-Jun-2020]. 2020. URL: https://mathworld.wolfram.com/JacobiPolynomial.html.

[96] Encyclopedia of Mathematics. *Favard theorem*. [Online; accessed 17-June-2020]. URL: http://encyclopediaofmath.org/index.php?title=Favard_theorem&oldid=43634.

[97] A. Wünsche. "Hermite and Laguerre 2D polynomials". In: *Journal of Computational and Applied Mathematics* 133.1 (2001). 5th Int. Symp. on Orthogonal Polynomials, Special Functions and their Applications, pp. 665–678. ISSN: 0377-0427. DOI: 10.1016/S0377-0427(00)00681-6.

[98] Yuan Xu. *Orthogonal polynomials of several variables*. 2017. arXiv: 1701.02709 [math.CA].

[99] Roberto Barrio, Juan Manuel Peña, and Tomas Sauer. "Three term recurrence for the evaluation of multivariate orthogonal polynomials". In: *Journal of Approximation Theory* 162.2 (2010), pp. 407–420. ISSN: 0021-9045. DOI: 10.1016/j.jat.2009.07.005.

[100] Zemax LLC. *OpticStudio 20.1 Release Notes*. https://my.zemax.com/en-US/OpticStudio-downloads/release-notes/. Jan. 2020.

[101] Philipp Jester, Christoph Menke, and Karsten Urban. "B-spline representation of optical surfaces and its accuracy in a ray trace algorithm". In: *Appl. Opt.* 50.6 (Feb. 2011), pp. 822–828. DOI: 10.1364/AO.50.000822.

[102] Holden Chase. "Optical design with rotationally symmetric NURBS". In: *International Optical Design Conference 2002*. Ed. by Paul K. Manhart and Jose M. Sasian. Vol. 4832. International Society for Optics and Photonics. SPIE, 2002, pp. 10–24. DOI: 10.1117/12.486490.

[103] Rengmao Wu et al. "Constructing optical freeform surfaces using unit tangent vectors of feature data points". In: *J. Opt. Soc. Am. A* 28.9 (Sept. 2011), pp. 1880–1888. DOI: 10.1364/JOSAA.28.001880.

[104] Shin-ya Morita et al. "Ray-tracing simulation method using piecewise quadratic interpolant for aspheric optical systems". In: *Appl. Opt.* 49.18 (June 2010), pp. 3442–3451. DOI: 10.1364/AO.49.003442.

[105] Martin D Buhmann. *Radial basis functions: theory and implementations*. Vol. 12. Cambridge university press, 2003.

[106] Christopher Stover. *Radial Function*. From MathWorld–A Wolfram Web Resource, created by Eric W. Weisstein [Online; accessed 25-Jun-2020]. 2020. URL: https://mathworld.wolfram.com/RadialFunction.html.

[107] Ozan Cakmakci et al. "Optimal local shape description for rotationally non-symmetric optical surface design and analysis". In: *Opt. Express* 16.3 (Feb. 2008), pp. 1583–1589. DOI: 10.1364/OE.16.001583.

[108] Ilhan Kaya and Jannick P. Rolland. "Hybrid Radial Basis Function and Local ϕ-Polynomial Surfaces". In: *Renewable Energy and the Environment: Postdeadline Papers*. Optical Society of America, 2013, FW2B.4. DOI: 10.1364/FREEFORM.2013.FW2B.4.

[109] Dmitry Reshidko and Jose Sasian. "Method for the design of nonaxially symmetric optical systems using free-form surfaces". In: *Optical Engineering* 57.10 (2018), pp. 1–10. DOI: 10.1117/1.OE.57.10.101704.

[110] M. V. Svechnikov et al. "Resolving capacity of the circular Zernike polynomials". In: *Opt. Express* 23.11 (June 2015), pp. 14677–14694. DOI: 10.1364/OE.23.014677.

[111] Ilhan Kaya, Kevin P. Thompson, and Jannick P. Rolland. "Comparative assessment of freeform polynomials as optical surface descriptions". In: *Opt. Express* 20.20 (Sept. 2012), pp. 22683–22691. DOI: 10.1364/OE.20.022683.

[112] Donglin Ma et al. "Freeform optics construction with nonuniformly sampled grids in modified double-pole coordinate system". In: *Optical Engineering* 54.12 (2015), pp. 1–8. DOI: 10.1117/1.OE.54.12.125102.

[113] Tong Yang, Guo-Fan Jin, and Jun Zhu. "Automated design of freeform imaging systems". In: *Light: Science & Applications* 6.10 (2017), e17081–e17081.

[114] Thibault Grillon et al. "Freeform geometrical optics II: from parametric representation to CAD/CAM". In: *Appl. Opt.* 58.34 (Dec. 2019), pp. 9465–9472. DOI: 10.1364/AO.58.009465.

[115] Jean-Baptiste Volatier and Guillaume Druart. "Differential method for freeform optics applied to two-mirror off-axis telescope design". In: *Opt. Lett.* 44.5 (Mar. 2019), pp. 1174–1177. DOI: 10.1364/OL.44.001174.

[116] Eduard Muslimov et al. "Combining freeform optics and curved detectors for wide field imaging: a polynomial approach over squared aperture". In: *Opt. Express* 25.13 (June 2017), pp. 14598–14610. DOI: 10.1364/OE.25.014598.

[117] Isaac Trumper et al. "Design form classification of two-mirror unobstructed freeform telescopes". In: *Optical Engineering* 59.2 (2020), pp. 1–15. DOI: 10.1117/1.OE.59.2.025105.

[118] R. F. Horton, Thom Peck, and Art Colgate. "All reflective 2 mirror unobscured wide field telescope/collimator designs". In: *Ground-based and Airborne Telescopes III*. Ed. by Larry M. Stepp, Roberto Gilmozzi, and Helen J. Hall. Vol. 7733. International Society for Optics and Photonics. SPIE, 2010, pp. 1024–1036. DOI: 10.1117/12.857310.

[119] Eduard Muslimov et al. "Design of optical systems with toroidal curved detectors". In: *Opt. Lett.* 43.13 (July 2018), pp. 3092–3095. DOI: 10.1364/OL.43.003092.

[120] Jannick P. Rolland et al. "Pamplemousse: The optical design, fabrication, and assembly of a three-mirror freeform imaging telescope". In: *International Optical Design Conference 2014*. Ed. by Mariana Figueiro et al. Vol. 9293. International Society for Optics and Photonics. SPIE, 2014, pp. 172–177. DOI: 10.1117/12.2074479.

[121] Jun Zhu et al. "Design of a low F-number freeform off-axis three-mirror system with rectangular field-of-view". In: *Journal of Optics* 17.1 (Dec. 2014), p. 015605. DOI: 10.1088/2040-8978/17/1/015605.

[122] Y. Zhong et al. "Investigation of TMA systems with different freeform surfaces". In: *Optical Systems Design 2015: Optical Design and Engineering VI*. Ed. by Laurent

Mazuray, Rolf Wartmann, and Andrew P. Wood. Vol. 9626. International Society for Optics and Photonics. SPIE, 2015, pp. 229–238. DOI: 10.1117/12.2191228.
[123] Mateusz Oleszko and Herbert Gross. "Analysis of freeform mirror systems based on the decomposition of the total wave aberration into Zernike surface contributions". In: *Appl. Opt.* 57.9 (Mar. 2018), pp. 1998–2006. DOI: 10.1364/AO.57.001998.
[124] Wei Hou et al. "Construction method through forward and reverse ray tracing for a design of ultra-wide linear field-of-view off-axis freeform imaging systems". In: *Journal of Optics* 17.5 (Apr. 2015), p. 055603. DOI: 10.1088/2040-8978/17/5/055603.
[125] Zexiao Li et al. "Integrated manufacture of a freeform off-axis multi-reflective imaging system without optical alignment". In: *Opt. Express* 26.6 (Mar. 2018), pp. 7625–7637. DOI: 10.1364/OE.26.007625.
[126] Qingyu Meng et al. "Design of off-axis three-mirror systems with ultrawide field of view based on an expansion process of surface freeform and field of view". In: *Appl. Opt.* 58.3 (Jan. 2019), pp. 609–615. DOI: 10.1364/AO.58.000609.
[127] Weichen Wu, Guofan Jin, and Jun Zhu. "Optical design of the freeform reflective imaging system with wide rectangular FOV and low F-number". In: *Results in Physics* 15 (2019), p. 102688. ISSN: 2211-3797. DOI: 10.1016/j.rinp.2019.102688.
[128] Luo Gu et al. "Design and fabrication of an off-axis four-mirror system for head-up displays". In: *Appl. Opt.* 59.16 (June 2020), pp. 4893–4900. DOI: 10.1364/AO.392602.
[129] Jonathan C. Papa, Joseph M. Howard, and Jannick P. Rolland. "Starting point designs for freeform four-mirror systems". In: *Optical Engineering* 57.10 (2018), pp. 1–11. DOI: 10.1117/1.OE.57.10.101705.
[130] Ruoxin Liu et al. "A Design for a Manufacturing-Constrained Off-Axis Four-Mirror Reflective System". In: *Applied Sciences* 10.15 (Aug. 2020), p. 5387. DOI: 10.3390/app10155387.
[131] Jonathan C. Papa, Joseph M. Howard, and Jannick P. Rolland. "Three-mirror freeform imagers". In: *Optical Design and Engineering VII*. Ed. by Laurent Mazuray, Rolf Wartmann, and Andrew P. Wood. Vol. 10690. International Society for Optics and Photonics. SPIE, 2018, pp. 312–326. DOI: 10.1117/12.2314403.
[132] Warren J. Smith. *Modern lens design*. second. McGraw-Hill, 2005. ISBN: 0071438300.
[133] Takayuki Nakano and Yasuhisa Tamagawa. "Configuration of an off-axis three-mirror system focused on compactness and brightness". In: *Appl. Opt.* 44.5 (Feb. 2005), pp. 776–783. DOI: 10.1364/AO.44.000776.
[134] Chen Xu et al. "Automatic optical path configuration variation in off-axis mirror system design". In: *Opt. Express* 27.11 (May 2019), pp. 15251–15261. DOI: 10.1364/OE.27.015251.
[135] Dmitry Reshidko, Masatsugu Nakanato, and José Sasián. "Ray Tracing Methods for Correcting Chromatic Aberrations in Imaging Systems". In: *International Journal of Optics* (Mar. 2014). DOI: 10.1155/2014/351584.
[136] John R. Rogers. "Techniques and tools for obtaining symmetrical performance from tilted-component systems". In: *Optical Engineering* 39.7 (2000), pp. 1776–1787. DOI: 10.1117/1.602557.
[137] Yi Zhong and Herbert Gross. "Initial system design method for non-rotationally symmetric systems based on Gaussian brackets and Nodal aberration theory". In: *Opt. Express* 25.9 (May 2017), pp. 10016–10030. DOI: 10.1364/OE.25.010016.
[138] Lucimara Cristina et al. "Two-mirror telescope design with third-order coma insensitive to decenter misalignment". In: *Opt. Express* 21.6 (Mar. 2013), pp. 6851–6865. DOI: 10.1364/OE.21.006851.
[139] Jun Zhu et al. "Generating optical freeform surfaces considering both coordinates and normals of discrete data points". In: *J. Opt. Soc. Am. A* 31.11 (Nov. 2014), pp. 2401–2408. DOI: 10.1364/JOSAA.31.002401.
[140] Pablo Gimenez-Benitez et al. "Simultaneous multiple surface optical design method in three dimensions". In: *Optical Engineering* 43.7 (2004), pp. 1489–1502. DOI: 10.1117/1.1752918.
[141] Jean-Baptiste Volatier, Álvaro Menduiña-Fernández, and Markus Erhard. "Generalization of differential ray tracing by automatic differentiation of computational graphs". In: *J. Opt. Soc. Am. A* 34.7 (July 2017), pp. 1146–1151. DOI: 10.1364/JOSAA.34.001146.
[142] Alvaro Menduina Fernandez. "Optimum design of freeform-enabled space optical instruments". PhD thesis. Delft University of Technology, 2017. URL: http://resolver.tudelft.nl/uuid:538fd43d-5ea8-41ca-8992-7f40f8abc450.
[143] Jean-Baptiste Volatier, Louis Duveau, and Guillaume Druart. "An exploration of the freeform two-mirror off-axis solution space". In: *Journal of Physics: Photonics* 2.1 (Dec. 2019), p. 014004. DOI: 10.1088/2515-7647/ab5c0d.

[144] Thomas Houllier and Thierry Lépine. "Comparing optimization algorithms for conventional and freeform optical design". In: *Opt. Express* 27.13 (June 2019), pp. 18940–18957. DOI: 10.1364/OE.27.018940.

[145] Malacara-Hernández Daniel and Malacara-Hernández Zacarias. *Handbook of optical design*. 3rd ed. CRC Press, 2013. ISBN: 978-1-4398-6801-0.

[146] G. W. Forbes. "Optical system assessment for design: numerical ray tracing in the Gaussian pupil". In: *J. Opt. Soc. Am. A* 5.11 (Nov. 1988), pp. 1943–1956. DOI: 10.1364/JOSAA.5.001943.

[147] Greg W. Forbes et al. "Ray selection for optimization of rotationally symmetric systems". In: *Advanced Optical Technologies* 5.3 (2016), pp. 237–247. DOI: 10.1515/aot-2016-0019.

[148] Eric W. Weisstein. *Logarithmic Spiral*. From MathWorld–A Wolfram Web Resource. 2020. URL: https://mathworld.wolfram.com/LogarithmicSpiral.html.

[149] Eric W. Weisstein. *Archimedean Spiral*. From MathWorld–A Wolfram Web Resource. 2020. URL: https://mathworld.wolfram.com/ArchimedeanSpiral.html.

[150] Richard Swinbank and R. James Purser. "Fibonacci grids: A novel approach to global modelling". In: *Quarterly Journal of the Royal Meteorological Society* 132.619 (2006), pp. 1769–1793. DOI: 10.1256/qj.05.227.

[151] Colas Schretter, Leif Kobbelt, and Paul-Olivier Dehaye. "Golden Ratio Sequences for Low-Discrepancy Sampling". In: *Journal of Graphics Tools* 16.2 (2012), pp. 95–104. DOI: 10.1080/2165347X.2012.679555.

[152] Riichirou Negishi and Kumiko Sekiguchi. "Pixel-Filling by Using Fibonacci Spiral". In: *FORMA* 22.3 (Dec. 2007), pp. 207–215. ISSN: 09116036. URL: https://ci.nii.ac.jp/naid/10026050998/en/.

[153] Ladislav Kocis and William J. Whiten. "Computational Investigations of Low-Discrepancy Sequences". In: *ACM Trans. Math. Softw.* 23.2 (June 1997), pp. 266–294. ISSN: 0098-3500. DOI: 10.1145/264029.264064.

[154] Eric W. Weisstein. *Disk Point Picking*. From MathWorld–A Wolfram Web Resource. 2020. URL: https://mathworld.wolfram.com/DiskPointPicking.html.

[155] Thouis R. Jones. "Efficient Generation of Poisson-Disk Sampling Patterns". In: *Journal of Graphics Tools* 11.2 (2006), pp. 27–36. DOI: 10.1080/2151237X.2006.10129217.

[156] Tony Hull et al. "Mid-spatial frequency matters: examples of the control of the power spectral density and what that means to the performance of imaging systems". In: *Infrared Technology and Applications XXXVIII*. Ed. by Bjørn F. Andresen, Gabor F. Fulop, and Paul R. Norton. Vol. 8353. International Society for Optics and Photonics. SPIE, 2012, pp. 799–815. DOI: 10.1117/12.921097.

[157] G. W. Forbes. "Never-ending struggles with mid-spatial frequencies". In: *Optical Measurement Systems for Industrial Inspection IX*. Ed. by Peter Lehmann, Wolfgang Osten, and Armando Albertazzi Gonçalves Jr. Vol. 9525. International Society for Optics and Photonics. SPIE, 2015, pp. 307–316. DOI: 10.1117/12.2191135.

[158] Thomas Houllier et al. "Advanced optical freeform substrates fabricated by ceramic 3D printing and controlled by deflectometry". In: *Optical Fabrication, Testing, and Metrology VI*. Ed. by Sven Schröder and Roland Geyl. Vol. 10692. International Society for Optics and Photonics. SPIE, 2018, pp. 171–182. DOI: 10.1117/12.2312649.

[159] M. Patti et al. "MAORY optical design analysis and tolerances". In: *Optical Design and Engineering VII*. Ed. by Laurent Mazuray, Rolf Wartmann, and Andrew P. Wood. Vol. 10690. International Society for Optics and Photonics. SPIE, 2018, pp. 483–497. DOI: 10.1117/12.2312067.

[160] Nick Takaki, Aaron Bauer, and Jannick P. Rolland. "On-the-fly surface manufacturability constraints for freeform optical design enabled by orthogonal polynomials". In: *Opt. Express* 27.5 (Mar. 2019), pp. 6129–6146. DOI: 10.1364/OE.27.006129.

[161] Richard N Pfisterer. "Approximated scatter models for stray light analysis". In: *Optics & Photonics News* 22.10 (2011), pp. 16–17.

[162] Alan W. Greynolds. "Scalability of a cross-platform multi-threaded non-sequential optical ray tracer". In: *Novel Optical Systems Design and Optimization XIV*. Ed. by R. John Koshel and G. Groot Gregory. Vol. 8129. International Society for Optics and Photonics. SPIE, 2011, pp. 103–112. DOI: 10.1117/12.891920.

[163] Steven G. Parker et al. "OptiX: A General Purpose Ray Tracing Engine". In: *ACM Trans. Graph.* 29.4 (July 2010). ISSN: 0730-0301. DOI: 10.1145/1778765.1778803.

[164] Florian Mauch et al. "Open-source graphics processing unit–accelerated ray tracer for optical simulation". In: *Optical Engineering* 52.5 (2013), pp. 1–6. DOI: 10.1117/1.OE.52.5.053004.

[165] Manuel Tessmer and Herbert Gross. "Generalized propagation of light through optical systems. II. Numerical implications". In: *J. Opt. Soc. Am. A* 32.12 (Dec. 2015), pp. 2276–2285. DOI: 10.1364/JOSAA.32.002276.

[166] Herbert Gross. *Handbook of Optical Systems, Volume 1, Fundamentals of Technical Optics*. Wiley-VCH, 2005. ISBN: 3-527-40377-9.

[167] G. H. Spencer and M. V. R. K. Murty. "General Ray-Tracing Procedure". In: *J. Opt. Soc. Am.* 52.6 (June 1962), pp. 672–678. DOI: 10.1364/JOSA.52.000672.

[168] James Eugene Klein. "Demystifying the sequential ray-tracing algorithm". In: *Optical Design and Analysis Software II*. Ed. by Richard C. Juergens. Vol. 4769. International Society for Optics and Photonics. SPIE, 2002, pp. 67–74. DOI: 10.1117/12.481191.

[169] Carsten C. Reichert, Tarik Gruhonjic, and Alois M. Herkommer. "Development of an open source algorithm for optical system design, combining genetic and local optimization". In: *Optical Engineering* 59.5 (2020), pp. 1–13. DOI: 10.1117/1.OE.59.5.055111.

[170] Neil Savage. "Optical Design Software". In: *Nature Photonics* 1 (10 2007), pp. 598–599. DOI: 10.1038/nphoton.2007.190.

[171] Eric W. Weisstein. *Normal Vector*. From MathWorld–A Wolfram Web Resource [Online; accessed 20-Oct-2020]. 2020. URL: https://mathworld.wolfram.com/NormalVector.html.

[172] Juan L. Rayces and Lan Lebich. "Modeling of diffractive optical elements for lens design". In: *Current Developments in Optical Design and Optical Engineering III*. Ed. by Robert E. Fischer and Warren J. Smith. Vol. 2000. International Society for Optics and Photonics. SPIE, 1993, pp. 43–52. DOI: 10.1117/12.163660.

[173] W.T. Welford. "VI Aberration Theory of Gratings and Grating Mountings". In: ed. by E. Wolf. Vol. 4. Progress in Optics. Elsevier, 1965, pp. 241–280. DOI: 10.1016/S0079-6638(08)70493-1.

[174] Jacob Reimers et al. "Freeform spectrometer enabling increased compactness". In: *Light: Science & Applications* 6 (7 Feb. 2017). DOI: 10.1038/lsa.2017.26.

[175] Scott Allen Lerner. *Optical design using novel aspheric surfaces*. 2000. URL: http://hdl.handle.net/10150/289160.

[176] *Steel Bank Common Lisp*. [Online; accessed 23-Oct-2020]. URL: http://sbcl.org/.

[177] Enrique Alba and José M. Troya. "A Survey of Parallel Distributed Genetic Algorithms". In: *Complexity* 4.4 (Mar. 1999), pp. 31–52. ISSN: 1076-2787. DOI: 10.1002/(SICI)1099-0526(199903/04)4:4%3C31::AID-CPLX5%3E3.0.CO;2-4.

[178] Soniya Lalwani et al. "A Survey on Parallel Particle Swarm Optimization Algorithms". In: *Arabian Journal for Science and Engineering* 44 (4 Apr. 2019), pp. 2899–2923. DOI: 10.1007/s13369-018-03713-6.

[179] Doug Hoyte. *Let Over Lambda*. 2008. ISBN: 978-1-4357-1275-1.

[180] Steffen Finck et al. *Real-parameter black-box optimization benchmarking 2010: Presentation of the noisy functions*. Tech. rep. Citeseer, 2010.

[181] Jörg Stork, Agoston E Eiben, and Thomas Bartz-Beielstein. "A new taxonomy of continuous global optimization algorithms". In: *arXiv preprint arXiv:1808.08818* (2018).

[182] Nikolaus Hansen et al. "Comparing Results of 31 Algorithms from the Black-Box Optimization Benchmarking BBOB-2009". In: *Proceedings of the 12th Annual Conference Companion on Genetic and Evolutionary Computation*. GECCO '10. Portland, Oregon, USA: Association for Computing Machinery, 2010, pp. 1689–1696. ISBN: 9781450300735. DOI: 10.1145/1830761.1830790.

[183] Wikipedia contributors. *Genetic and Evolutionary Computation Conference — Wikipedia, The Free Encyclopedia*. https://en.wikipedia.org/w/index.php?title=Genetic_and_Evolutionary_Computation_Conference&oldid=794822432. [Online; accessed 2-September-2020]. 2017.

[184] J. A. Nelder and R. Mead. "A Simplex Method for Function Minimization". In: *The Computer Journal* 7.4 (Jan. 1965), pp. 308–313. ISSN: 0010-4620. DOI: 10.1093/comjnl/7.4.308.

[185] M.A. Luersen, R. Le Riche, and F. Guyon. "A constrained, globalized, and bounded Nelder–Mead method for engineering optimization". In: *Structural and Multidisciplinary Optimization* 27 (1 May 2004), pp. 43–54. DOI: 10.1007/s00158-003-0320-9.

[186] Thomas Houllier and Thierry Lépine. "Search algorithms and optical systems design". In: Paris: Zemax ENVISION2019 (private non-indexed conference), Mar. 2019. DOI: 10.13140/RG.2.2.31951.12969.

[187] Wikipedia contributors. *Himmelblau's function — Wikipedia, The Free Encyclopedia*. https://en.wikipedia.org/w/index.php?title=Himmelblau%27s_function&oldid=961144451. [Online; accessed 2-September-2020]. 2020.

[188] Thomas Houllier. *simplex.wmv*. https://www.researchgate.net/publication/333916933_simplexwmv. [Online]. 2019.

[189] Thomas Houllier. *gsa.wmv*. https://www.researchgate.net/publication/333916932_gsawmv. [Online]. 2019.
[190] Thomas Houllier. *pso.wmv*. https://www.researchgate.net/publication/333916848_psowmv. [Online]. 2019.
[191] Thomas Houllier. *cmaes.wmv*. https://www.researchgate.net/publication/333916767_cmaeswmv. [Online]. 2019.
[192] Thomas Houllier. *cs.wmv*. https://www.researchgate.net/publication/333916766_cswmv. [Online]. 2019.
[193] Maxim Sakharov, Thomas Houllier, and Thierry Lépine. "Mind Evolutionary Computation Co-algorithm for Optimizing Optical Systems". In: *Proceedings of the Fourth International Scientific Conference "Intelligent Information Technologies for Industry" (IITI'19)*. Ed. by Sergey Kovalev et al. Cham: Springer International Publishing, 2020, pp. 476–486. ISBN: 978-3-030-50097-9. DOI: 10.1007/978-3-030-50097-9_48.
[194] Thomas Houllier and Thierry Lépine. "Optical design and optimization strategies: state of the art and future developments". In: Nice, France: Workshop on Machine-Learning-Assisted Image Formation, July 2019. DOI: 10.13140/RG.2.2.28749.87524.
[195] Marco Dorigo and Christian Blum. "Ant colony optimization theory: A survey". In: *Theoretical Computer Science* 344.2 (2005), pp. 243–278. ISSN: 0304-3975. DOI: 10.1016/j.tcs.2005.05.020.
[196] Ziyao Tang, Matthias Sonntag, and Herbert Gross. "Ant colony optimization in lens design". In: *Appl. Opt.* 58.23 (Aug. 2019), pp. 6357–6364. DOI: 10.1364/AO.58.006357.
[197] Wikipedia contributors. *Markov decision process — Wikipedia, The Free Encyclopedia*. https://en.wikipedia.org/w/index.php?title=Markov_decision_process&oldid=975754122. [Online; accessed 4-September-2020]. 2020.
[198] Yaochu Jin. "Surrogate-assisted evolutionary computation: Recent advances and future challenges". In: *Swarm and Evolutionary Computation* 1.2 (2011), pp. 61–70. ISSN: 2210-6502. DOI: 10.1016/j.swevo.2011.05.001.
[199] Kenneth E. Moore. "Optimization for as-built performance". In: *Photonic Instrumentation Engineering VI*. Ed. by Yakov G. Soskind. Vol. 10925. International Society for Optics and Photonics. SPIE, 2019, pp. 1–8. DOI: 10.1117/12.2508062.
[200] Tong Yang, Dewen Cheng, and Yongtian Wang. "Direct generation of starting points for freeform off-axis three-mirror imaging system design using neural network based deep-learning". In: *Opt. Express* 27.12 (June 2019), pp. 17228–17238. DOI: 10.1364/OE.27.017228.
[201] Geoffroi Côté, Jean-François Lalonde, and Simon Thibault. "Extrapolating from lens design databases using deep learning". In: *Opt. Express* 27.20 (Sept. 2019), pp. 28279–28292. DOI: 10.1364/OE.27.028279.
[202] Carlos A. Coello Coello et al. "Evolutionary multiobjective optimization: open research areas and some challenges lying ahead". In: *Complex & Intelligent Systems* 6 (June 2019), pp. 221–236. DOI: 10.1007/s40747-019-0113-4.
[203] Isaac Trumper, Buell T. Jannuzi, and Dae Wook Kim. "Emerging technology for astronomical optics metrology". In: *Optics and Lasers in Engineering* 104 (2018). Optical Tools for Metrology, Imaging and Diagnostics, pp. 22–31. ISSN: 0143-8166. DOI: 10.1016/j.optlaseng.2017.09.009.
[204] Shanyong Chen et al. "Measurement of Freeform Optical Surfaces: Trade-Off between Accuracy and Dynamic Range". In: *Laser & Photonics Reviews* 14.5 (2020), p. 1900365. DOI: 10.1002/lpor.201900365.
[205] Lei Huang et al. "Comparison of two-dimensional integration methods for shape reconstruction from gradient data". In: *Optics and Lasers in Engineering* 64 (2015), pp. 1–11. ISSN: 0143-8166. DOI: 10.1016/j.optlaseng.2014.07.002.
[206] Pierre Bon, Serge Monneret, and Benoit Wattellier. "Noniterative boundary-artifact-free wavefront reconstruction from its derivatives". In: *Appl. Opt.* 51.23 (Aug. 2012), pp. 5698–5704. DOI: 10.1364/AO.51.005698.
[207] Byoung Chang Kim et al. "Aspheric surface reconstruction from curvature data along two orthogonal directions". In: *Optifab 2015*. Ed. by Julie L. Bentley and Sebastian Stoebenau. Vol. 9633. International Society for Optics and Photonics. SPIE, 2015, pp. 640–647. DOI: 10.1117/12.2196063.
[208] P Rose, Y Surrel, and J M Becker. "Specific design requirements for a reliable slope and curvature measurement standard". In: *Measurement Science and Technology* 20.9 (Aug. 2009), p. 095110. DOI: 10.1088/0957-0233/20/9/095110.
[209] Shanyong Chen et al. "Subaperture stitching algorithms: A comparison". In: *Optics Communications* 390 (2017), pp. 61–71. ISSN: 0030-4018. DOI: 10.1016/j.optcom.2016.12.067.

[210] R. Henselmans et al. "The NANOMEFOS non-contact measurement machine for freeform optics". In: *Precision Engineering* 35.4 (2011), pp. 607–624. ISSN: 0141-6359. DOI: 10.1016/j.precisioneng.2011.04.004.

[211] Junqiang Ye et al. "In-situ deflectometic measurement of transparent optics in precision robotic polishing". In: *Precision Engineering* 64 (2020), pp. 63–69. ISSN: 0141-6359. DOI: 10.1016/j.precisioneng.2020.03.011.

[212] Xueyang Xu et al. "Extra-detection-free monoscopic deflectometry for the in situ measurement of freeform specular surfaces". In: *Opt. Lett.* 44.17 (Sept. 2019), pp. 4271–4274. DOI: 10.1364/OL.44.004271.

[213] Jianing Yao, Alexander Anderson, and Jannick P. Rolland. "Point-cloud noncontact metrology of freeform optical surfaces". In: *Opt. Express* 26.8 (Apr. 2018), pp. 10242–10265. DOI: 10.1364/OE.26.010242.

[214] Di Xu et al. "Cascade optical coherence tomography (C-OCT)". In: *Optics Express* 28.14 (June 2020), p. 19937. DOI: 10.1364/oe.394638.

[215] Di Xu et al. "Verification of cascade optical coherence tomography for freeform optics form metrology". In: *Optics Express* 29.6 (Mar. 2021), p. 8542. DOI: 10.1364/oe.413844.

[216] W Lyda et al. "Advantages of chromatic-confocal spectral interferometry in comparison to chromatic confocal microscopy". In: *Measurement Science and Technology* 23.5 (Mar. 2012), p. 054009. DOI: 10.1088/0957-0233/23/5/054009.

[217] STIL. [Online; accessed 17-September-2020]. URL: http://point.stil-sensors.com/specseverest.php?lang=EN.

[218] Dong-Hyeok Lee and Nahm-Gyoo Cho. "Assessment of surface profile data acquired by a stylus profilometer". In: *Measurement Science and Technology* 23.10 (Aug. 2012), p. 105601. DOI: 10.1088/0957-0233/23/10/105601.

[219] Taylor Hobson. *Form Talysurf® CNC Series*. Online; accessed 18-September-2020. URL: https://www.taylor-hobson.com/products/surface-profilers/form-talysurf-cnc-series/form-talysurf-cnc-series.

[220] Pauline Rose. "Développement de la déflectométrie à décalage de phase en contexte industriel : mesure de forme, étalonnage et normalisation". Written in French. PhD thesis. Université Jean Monnet, Saint-Étienne, 2011.

[221] Michael Schulz. "Topography measurement by a reliable large-area curvature sensor". In: *Optik* 112.2 (2001), pp. 86–90. ISSN: 0030-4026. DOI: 10.1078/0030-4026-00015.

[222] Ulf Griesmann et al. "Uncertainties in aspheric profile measurements with the geometry measuring machine at NIST". In: *Advanced Characterization Techniques for Optics, Semiconductors, and Nanotechnologies II*. Ed. by Angela Duparre, Bhanwar Singh, and Zu-Han Gu. Vol. 5878. International Society for Optics and Photonics. SPIE, 2005, pp. 112–124. DOI: 10.1117/12.615398.

[223] Nadia Machkour-Deshayes et al. "Form-Profiling of Optics Using the Geometry Measuring Machine and the M-48 CMM at NIST". In: *J Res Natl Inst Stand Technol.* 111 (5 Oct. 2006), pp. 373–384. DOI: 10.6028/jres.111.027.

[224] Dae Wook Kim et al. "Algorithms for surface reconstruction from curvature data for freeform aspherics". In: *Optical Manufacturing and Testing X*. Ed. by Oliver W. Fähnle, Ray Williamson, and Dae Wook Kim. Vol. 8838. International Society for Optics and Photonics. SPIE, 2013, pp. 60–68. DOI: 10.1117/12.2024285.

[225] SeongWon Lee et al. "Subaperture method for aspheric surface metrology using curvature data". In: *Optical Micro- and Nanometrology VI*. Ed. by Christophe Gorecki, Anand Krishna Asundi, and Wolfgang Osten. Vol. 9890. International Society for Optics and Photonics. SPIE, 2016, pp. 281–288. DOI: 10.1117/12.2227961.

[226] Weiyao Zou, Kevin P. Thompson, and Jannick P. Rolland. "Differential Shack-Hartmann curvature sensor: local principal curvature measurements". In: *Journal of the Optical Society of America A* 25.9 (Aug. 2008), p. 2331. DOI: 10.1364/josaa.25.002331.

[227] Ping Zhou and James H. Burge. "Fabrication error analysis and experimental demonstration for computer-generated holograms". In: *Appl. Opt.* 46.5 (Feb. 2007), pp. 657–663. DOI: 10.1364/AO.46.000657.

[228] Chunyu Zhao and James H. Burge. "Optical testing with computer generated holograms: comprehensive error analysis". In: *Optical Manufacturing and Testing X*. Ed. by Oliver W. Fähnle, Ray Williamson, and Dae Wook Kim. Vol. 8838. International Society for Optics and Photonics. SPIE, 2013, pp. 119–130. DOI: 10.1117/12.2024742.

[229] Ya Huang et al. "Computer generated holograms design in null testing of optical freeform surfaces". In: *International Conference on Optics in Precision Engineering and Nanotechnology (icOPEN2013)*. Ed. by Chenggen Quan, Kemao Qian, and Anand

Asundi. Vol. 8769. International Society for Optics and Photonics. SPIE, 2013, pp. 802–809. DOI: 10.1117/12.2021053.
[230] Shanyong Chen et al. "Reconfigurable optical null based on counter-rotating Zernike plates for test of aspheres". In: *Opt. Express* 22.2 (Jan. 2014), pp. 1381–1386. DOI: 10.1364/OE.22.001381.
[231] Kyle Fuerschbach, Kevin P. Thompson, and Jannick P. Rolland. "Interferometric measurement of a concave, φ-polynomial, Zernike mirror". In: *Opt. Lett.* 39.1 (Jan. 2014), pp. 18–21. DOI: 10.1364/OL.39.000018.
[232] Shuai Xue et al. "Adaptive null interferometric test using spatial light modulator for free-form surfaces". In: *Opt. Express* 27.6 (Mar. 2019), pp. 8414–8428. DOI: 10.1364/OE.27.008414.
[233] Lei Zhang et al. "Pure adaptive interferometer for free form surfaces metrology". In: *Opt. Express* 26.7 (Apr. 2018), pp. 7888–7898. DOI: 10.1364/OE.26.007888.
[234] Shijie Liu et al. "Study on measurement accuracy of active optics null test systems based on liquid crystal spatial light modulator and laser interferometer". In: *Optical Measurement Systems for Industrial Inspection X*. Ed. by Peter Lehmann, Wolfgang Osten, and Armando Albertazzi Gonçalves Jr. Vol. 10329. International Society for Optics and Photonics. SPIE, 2017, pp. 42–54. DOI: 10.1117/12.2269435.
[235] Romita Chaudhuri, Jonathan Papa, and Jannick P. Rolland. "System design of a single-shot reconfigurable null test using a spatial light modulator for freeform metrology". In: *Optics Letters* 44.8 (Apr. 2019), p. 2000. DOI: 10.1364/ol.44.002000.
[236] Young-Sik Ghim et al. "3D surface mapping of freeform optics using wavelength scanning lateral shearing interferometry". In: *Opt. Express* 22.5 (Mar. 2014), pp. 5098–5105. DOI: 10.1364/OE.22.005098.
[237] Sabrina Velghe et al. "Wave-front reconstruction from multidirectional phase derivatives generated by multilateral shearing interferometers". In: *Opt. Lett.* 30.3 (Feb. 2005), pp. 245–247. DOI: 10.1364/OL.30.000245.
[238] Daniel R. Neal, James Copland, and David A. Neal. "Shack-Hartmann wavefront sensor precision and accuracy". In: *Advanced Characterization Techniques for Optical, Semiconductor, and Data Storage Components*. Ed. by Angela Duparré and Bhanwar Singh. Vol. 4779. International Society for Optics and Photonics. SPIE, 2002, pp. 148–160. DOI: 10.1117/12.450850.
[239] Daniel M. Topa. "Wavefront reconstruction for the Shack-Hartmann wavefront sensor". In: *Optical Design and Analysis Software II*. Ed. by Richard C. Juergens. Vol. 4769. International Society for Optics and Photonics. SPIE, 2002, pp. 101–115. DOI: 10.1117/12.481179.
[240] J. Pfund, N. Lindlein, and J. Schwider. "Dynamic range expansion of a Shack–Hartmann sensor by use of a modified unwrapping algorithm". In: *Opt. Lett.* 23.13 (July 1998), pp. 995–997. DOI: 10.1364/OL.23.000995.
[241] Linda Lundström and Peter Unsbo. "Unwrapping Hartmann-Shack Images from Highly Aberrated Eyes Using an Iterative B-spline Based Extrapolation Method". In: *Optometry and Vision Science* 81 (5 May 2004), pp. 383–388. DOI: 10.1097/01.opx.0000135086.61760.b7.
[242] Vasyl V. Molebny. "Scanning Shack-Hartmann wavefront sensor". In: *Laser Radar Technology and Applications IX*. Ed. by Gary W. Kamerman and Gary W. Kamerman. Vol. 5412. International Society for Optics and Photonics. SPIE, 2004, pp. 66–71. DOI: 10.1117/12.541755.
[243] Imagine Optic. *HASO4 First Specifications*. [Online; accessed 18-September-2020]. URL: https://www.imagine-optic.com/product/haso4-first/#specifications.
[244] Zonghua Zhang et al. "Three-Dimensional Shape Measurements of Specular Objects Using Phase-Measuring Deflectometry". In: *Sensors* 17.12 (Dec. 2017), p. 2835. ISSN: 1424-8220. DOI: 10.3390/s17122835.
[245] Lei Huang et al. "Review of phase measuring deflectometry". In: *Optics and Lasers in Engineering* 107 (2018), pp. 247–257. ISSN: 0143-8166. DOI: 10.1016/j.optlaseng.2018.03.026.
[246] Yongjia Xu, Feng Gao, and Xiangqian Jiang. "A brief review of the technological advancements of phase measuring deflectometry". In: *PhotoniX* 1, 14 (1 June 2020). DOI: 10.1186/s43074-020-00015-9.
[247] Ali Pouya Fard. "Low uncertainty surface area measurement using deflectometry". PhD thesis. The University of North Carolina at Charlotte, 2018.
[248] Logan Rodriguez Graves. "Freeform Metrology Using Deflectometry". PhD thesis. The University of Arizona, 2019.
[249] Guillaume Butel. "Analysis and New Developments Towards Reliable and Portable Measurements in Deflectometry". PhD thesis. The University of Arizona, 2017.

[250] Run Huang. "High precision optical surface metrology using deflectometry". PhD thesis. The University of Arizona, 2015.

[251] Jonas Bartsch et al. "Effects of non-ideal display properties in phase measuring deflectometry: A model-based investigation". In: *Optical Micro- and Nanometrology VII*. Ed. by Christophe Gorecki, Anand Krishna Asundi, and Wolfgang Osten. Vol. 10678. International Society for Optics and Photonics. SPIE, 2018, pp. 192–201. DOI: 10.1117/12.2306463.

[252] D. Maestro-Watson, A. Izaguirre, and N. Arana-Arexolaleiba. "LCD screen calibration for deflectometric systems considering a single layer refraction model". In: *2017 IEEE International Workshop of Electronics, Control, Measurement, Signals and their Application to Mechatronics (ECMSM)*. 2017, pp. 1–6. DOI: 10.1109/ECMSM.2017.7945890.

[253] Peng Su et al. "SCOTS: a reverse Hartmann test with high dynamic range for Giant Magellan Telescope primary mirror segments". In: *Modern Technologies in Space- and Ground-based Telescopes and Instrumentation II*. Ed. by Ramón Navarro, Colin R. Cunningham, and Eric Prieto. Vol. 8450. International Society for Optics and Photonics. SPIE, 2012, pp. 332–340. DOI: 10.1117/12.926719.

[254] J. H. Burge et al. "Measuring large mirrors using SCOTS: the Software Configurable Optical Test System". In: *Advances in Optical and Mechanical Technologies for Telescopes and Instrumentation*. Ed. by Ramón Navarro, Colin R. Cunningham, and Allison A. Barto. Vol. 9151. International Society for Optics and Photonics. SPIE, 2014, pp. 311–323. DOI: 10.1117/12.2057726.

[255] Florian Willomitzer et al. "Hand-guided qualitative deflectometry with a mobile device". In: *Opt. Express* 28.7 (Mar. 2020), pp. 9027–9038. DOI: 10.1364/OE.383475.

[256] Lili Jiang et al. "Wavefront aberration metrology based on transmitted fringe deflectometry". In: *Appl. Opt.* 56.26 (Sept. 2017), pp. 7396–7403. DOI: 10.1364/AO.56.007396.

[257] Caixia Chang et al. "Measurement of the Three-Dimensional Shape of Discontinuous Specular Objects Using Infrared Phase-Measuring Deflectometry". In: *Sensors* 19.21 (Oct. 2019), p. 4621. ISSN: 1424-8220. DOI: 10.3390/s19214621.

[258] L. R. Graves et al. "Infinite deflectometry enabling 2π-steradian measurement range". In: *Opt. Express* 27.5 (Mar. 2019), pp. 7602–7615. DOI: 10.1364/OE.27.007602.

[259] Yongjia Xu et al. "A holistic calibration method with iterative distortion compensation for stereo deflectometry". In: *Optics and Lasers in Engineering* 106 (2018), pp. 111–118. ISSN: 0143-8166. DOI: 10.1016/j.optlaseng.2018.02.018.

[260] Xiangcheng Chen et al. "Color-coding and phase-shift method for absolute phase measurement". In: *Optics Communications* 298-299 (2013), pp. 54–58. ISSN: 0030-4018. DOI: 10.1016/j.optcom.2013.02.013.

[261] Jorge L. Flores et al. "Binary coded triangular fringes for 3-D surface-shape measurement". In: *Appl. Opt.* 52.15 (May 2013), pp. 3576–3582. DOI: 10.1364/AO.52.003576.

[262] Song Zhang. "Composite phase-shifting algorithm for absolute phase measurement". In: *Optics and Lasers in Engineering* 50.11 (2012), pp. 1538–1541. ISSN: 0143-8166. DOI: 10.1016/j.optlaseng.2012.06.005.

[263] Quentin Turchette and Trey Turner. "Inspection of Optics using ISO Specifications". In: *Optics & Photonics News* (2012).

[264] Fabian Etzold et al. "A novel approach towards standardizing surface quality inspection". In: *Third European Seminar on Precision Optics Manufacturing*. Ed. by Rolf Rascher et al. Vol. 10009. International Society for Optics and Photonics. SPIE, 2016, pp. 51–60. DOI: 10.1117/12.2235743.

[265] Thomas Houllier et al. "Measurement of a freeform mirror with strong altitude and slope departures using deflectometry". In: (2021). Sent to a journal and rejected. In the process of being rewritten. A draft of this article is included in the body of the present manuscript.

[266] Max Born et al. *Principles of Optics*. 7th expanded. Cambridge University Press, Cambridge, UK, 1999. ISBN: 0521642221.

[267] Robert J. Noll. "Zernike polynomials and atmospheric turbulence". In: *J. Opt. Soc. Am.* 66.3 (Mar. 1976), pp. 207–211. DOI: 10.1364/JOSA.66.000207.

[268] *Optics and photonics — Preparation of drawings for optical elements and systems — Part 5: Surface form tolerances*. Standard. International Organization for Standardization, Aug. 2015. URL: https://www.iso.org/standard/57566.html.

[269] Philip C. L. Stephenson. "Recurrence relations for the Cartesian derivatives of the Zernike polynomials". In: *J. Opt. Soc. Am. A* 31.4 (Apr. 2014), pp. 708–715. DOI: 10.1364/JOSAA.31.000708.

[270] Pavel Novák, Jiří Novák, and Antonín Mikš. "Fast and robust computation of Cartesian derivatives of Zernike polynomials". In: *Optics and Lasers in Engineering* 52 (2013), pp. 7–12. DOI: 10.1016/j.optlaseng.2013.07.012.

[271] Chee-Way Chong, P. Raveendran, and R. Mukundan. "A comparative analysis of algorithms for fast computation of Zernike moments". In: *Pattern Recognition* 36.3 (2003), pp. 731–742. DOI: 10.1016/S0031-3203(02)00091-2.