This work is a preprint of the paper Cruz, N. C., Monterreal, R., Redondo, J. L., Fernández-Reche, J., Enrique, R., & Ortigosa, P. M. (2022). Optical characterization of heliostat facets based on Computational Optimization. *Solar Energy*, 248, 1-15. There are authors from the University of Granada, the Solar Platform of Almería, and the University of Almería

DOI: https://doi.org/10.1016/j.solener.2022.10.043

It is published in Solar Energy (see: https://www.sciencedirect.com/science/article/abs/pii/S0038092X22007824 to access the published work).

Optical characterization of heliostat facets based on Computational Optimization

N.C. Cruz[a], R. Monterreal[c,*], J.L. Redondo[b], J. Fernández-Reche[c], R. Enrique[c], P.M. Ortigosa[b]

[a]*Dept. of Computer Architecture and Technology, University of Granada. Journalist Daniel Saucedo Street, 18014, Granada, Spain*
[b]*Dept. of Informatics, University of Almería, ceiA3 Excellence Agri-food Campus. Sacramento Road, 04120, Almería, Spain*
[c]*CIEMAT-PSA. Senés Road, km. 4.5, 04200, Tabernas, Almería, Spain*

Abstract

In solar power tower plants, knowing the optical quality of heliostats makes it possible to predict relevant information on the receiver surface, such as the irradiance concentration factor and spillage. However, there are no standardized routines for optical characterization in commercial facilities because the process is challenging, multidisciplinary, and time-demanding. This article revises the traditional optical characterization methodology followed at the Solar Platform of Almería (PSA). The process starts with the acquisition of the image of the studied optical system. After that, the picture must be fitted to an analytical model, which requires finding the variables that best reproduce the reality. The traditional method for accomplishing this task is iterative, semi-automatic, and contains trial-and-error components. This work studies how to replace this part with heuristic optimizers and considers using the state-of-the-art methods TLBO, UEGO, and Multi-Start Interior-Point (MSIP). Their effectiveness has been compared to the results manually achieved by an expert with three different heliostat facets. According to the results obtained, the parameter sets found by TLBO and UEGO outperform those obtained through the traditional method.

Keywords: Solar Power Tower Plants, Optical Characterization, Computational Optimization, Numerical Methods

1. Introduction

The increase in the world population and its demands require replacing the traditional and polluting energy sources with renewable and clean ones (Gallego and Camacho, 2018; Sah et al., 2020). Since solar energy the most abundant option (Gallego and Camacho, 2018), there is great interest in its exploitation (Kabir et al., 2018; Reddy et al., 2013). Among the existing technologies for

[1]Corresponding author: rafael.monterreal@psa.es

this purpose (Kabir et al., 2018; Reddy et al., 2013), concentrated solar power systems are promising because of their compatibility with hybridization, large-scale production, and storage capabilities (Reddy et al., 2013; Saghafifar et al., 2019). Solar Power Tower Plants (SPTP) stand out from this kind of system because of their higher efficiency and lower Levelized Cost of Electricity (LCOE) (de la Calle et al., 2020).

A typical SPTP consists of multiple sun-tracking mirrors, known as heliostats, which dynamically concentrate solar radiation on a receiver. The latter transfers the energy to a heat-transfer fluid used to generate electricity in a power cycle (Cruz et al., 2020; Reddy et al., 2013). Despite their conceptual simplicity, heliostats are not simple mirrors. They are complex systems consisting of multiple facets and have an impact on the performance of facilities (Fernández-Reche, 2006). For this reason, it is convenient to know the optical quality of heliostats, which is known as their Optical Characterization (OC in what follows). This information, coupled with image simulation models (Cruz et al., 2017), makes it possible to predict valuable data on the receiver. Some examples are the irradiance concentration factor, the image orientation and elongation, the image radius containing a certain percentage of the incoming power, and spillage (García et al., 2015). It is also valuable to detect malfunctions in heliostats that could reduce the throughput of the plant.

The process of OC consists in determining the set of parameters that best describe the behavior of the system, e.g., a complete heliostat or one of its facets. Thus, it is necessary to compare the real image of the characterized system to different synthetic replicas generated with an optical model and the appropriate parameters, e.g., the focal length for concave facets. The parameters that produce the most realistic synthetic image will define the OC. However, this procedure is time-demanding and challenging because it requires knowledge of image acquisition and analysis methods. For these reasons, there are no periodic OC programs in commercial SPTP, and this kind of task is generally carried out in heliostat prototypes at applied research centers, such as the *Plataforma Solar de Almería* (Spanish for Solar Platform of Almería) (PSA) in Spain.

For a commercial plant, it would be advisable to perform an OC campaign at several critical moments. The first one would be at the commissioning of the plant, i.e., before starting its regular operation, to confirm that it fulfills the specifications. Secondly, the solstices would be interesting because it is when the sun is at extreme positions. Thirdly, at the design point of the plant, which is likely to be one of the equinoxes. Fourthly, if the plant location has extreme temperatures, it would be advisable to test the optical quality with the highest and lowest temperatures to check their impact on the field. These tests would let us know the plant in detail. After that, the aging of the heliostats and their materials would determine further necessities, e.g., every two years, to ensure that the field is capable enough.

Despite not being an actively studied subject, there are several techniques in the literature for assessing the optical quality. In the 1980s, the intensity distribution reflected by heliostats was estimated by convolving several distributions coming from independent error sources. The fundamental ones were the sun shape, mirror waviness, and astigmatism. In this context, the error analysis was performed in the Fourier space (Kiera, 1985). In the 1990s, Sandia National Laboratories carried out different optical characterization campaigns of heliostats (Jones et al., 1995; Strachan and Houser, 1993). They used the BCS measurement system (Strachan, 1993) for real image acquisition and the HELIOS simulation software (Biggs and Vittitoe, 1979) for estimating the images produced under different parametric configurations, but the parameters to try were manually supervised by a human expert. Likewise, at the PSA, heliostat characterization has been performed since the 1990s to the present (Cordes, 1994; Monterreal, 1997) with the ProHERMES measurement system and the Fiat_Lux (Monterreal, 1999) heliostat image simulator. However, all these techniques share the handicap of heavily relying on specialized staff and considerable dedication to direct the parametric search and obtain reliable results.

In general, optical quality assessment techniques can be divided into two main classes: Indirect and Direct methods. The former studies the image of the sun reflected by the surface of the heliostat with an image analysis target and a measurement system. The most relevant parameters concerning the optical properties of the concentrator are then computed from the experimental data with different strategies. For instance, after modeling the heliostat in a proper simulator (King, 1982; Strachan and Houser, 1993), the model can be used along with the data to unfold an error distribution characterizing the non-deterministic slope errors of the heliostat. It is done by iterating through the dispersion (standard deviation) of a circular normal distribution in the simulator until the shape of the measured and predicted beams match (King, 1982; Strachan and Houser, 1993). It is also possible to compute the radiation intercepted by the receiver of a real concentrator by the convolution of the angular acceptance function, the optical error distribution, and the angular brightness of the radiation source. The parametrization of the theoretical distribution function makes it possible to iteratively seek the model parameters that best replicate the real image (Avellaner, 1980; Bendt and Rabl, 1981; Kiera and Schiel, 1989). Regarding Direct methods, they study either the heliostat surface or an image reflected on it, but from an artificial light source instead of the sun. It groups the most recent research lines in optical characterization (Kammel, 2003; Monterreal et al., 2017; Pottler et al., 2005).

The methodology currently followed at the PSA falls into the group of Indirect methods. The source of light reflected on the target directly comes from the sun. It is taken with a high-resolution digital camera at the characterization instant. The theoretical parametric model of the optical quality of the heliostat combines both variables of statistical nature, such as the slope error, and deterministic ones, such as the focal distance. King (1982); Strachan and Houser (1993) used the HELIOS software package, while Fiat_Lux is the choice at the PSA (Cruz et al., 2017). For its simulations, HELIOS follows a cone optics approach in which the flux density results from combining the error cone of the reflected rays and the sun shape by convolving independent distributions with the Fourier transform (Garcia et al., 2008). For its part, Fiat_Lux simulates the flux distributions produced by mirrors using normally-distributed random values of slope error and computing the trajectories for the bundle of reflected rays coming from the solar disk according to geometric optics laws. The latter is done for each unitary normal vector to the reflective surface (Garcia et al., 2008; Monterreal, 1999). Fiat_Lux also stands out due to its capability of taking as input a real image of the sun experimentally acquired, while other tools rely on synthetic images of the sun.

Regardless, the OC methodology is still semi-automatic due to the required parametric search. It relies on an expert to guide the definition of parameter sets to compare to real images (Monterreal, 1997). Although this process has been successfully applied for years, it is significantly time-demanding, and the explored parameter sets are inherently biased by the expert's vision. This work aims to update this method by approaching OC as an optimization problem (Nocedal and Wright, 2006; Törn and Žilinskas, 1989) focused on finding the set of parameters that results in the most realistic synthetic image. Therefore, this work continues the research line of modernization recently opened by Monterreal et al. (2022) and proposes applying computational optimization methods to replace the manual part of the OC strategy of the PSA.

The use of global optimization algorithms is expected to improve the quality of solutions and simplify their obtaining. Since the OC workflow involves applying non-linear models with different accuracy degrees and stochasticity, the optimizers considered are heuristics (Salhi, 2017). This sort of optimization algorithm relies on intuitive ideas and randomness rather than on rigorous certainty. They are known to provide a trade-off between computational effort and the quality of solutions. Namely, the optimizers studied in this work as a replacement of the manual work of experts are the following: i) TLBO (Rao et al., 2012), ii) UEGO (Jelasity et al., 2001), and iii) Multi-start Interior-Point (MSIP) (Griva et al., 2009; Salhi, 2017). Their results have been compared to those obtained through the traditional method of the PSA with three different heliostat facets.

The remainder of the paper has the following structure: Section 2 describes the materials and methods considered in this work, including the experimental setup and the OC strategies. After that, Section 3 shows the results obtained with each end method and the three facets considered. Finally, Section 4 contains the conclusions and states the future work.

2. Materials and methods

As introduced, OC is a multidisciplinary process covering several fields, from image acquisition to optical simulation and parameter fitting. These tasks rely on the use of specific hardware and software. This section describes the environment and equipment for OC at the PSA, both for the traditional approach and the new one based on Computational Optimization.

2.1. Image acquisition and simulation tools

The traditional OC methodology followed at the PSA has been applied with heliostat prototypes and facets since the nineties. It obtains the information about the studied optical system by comparing the sun image that it reflects on a target to the synthetic one generated by the appropriate models (Cruz et al., 2017). The revision of the method proposed in this work also inherits this background. Therefore, image acquisition is highly relevant for OC.

Roughly speaking, the images are obtained by making the optical system reflect the sun image onto a target in the tower of the CESA-I heliostat field (Cruz et al., 2018; Gallego and Camacho, 2018). Figure 1 provides a general overview of the referred facility and the elements involved in this work, including the three types of facets to be characterized. More precisely, the whole system is known as ProHERMES 2A (Monterreal and Neumann, 1994), and it consists of the following hardware and software components:

- FUJINON LENS. Model: H14X10.5A-R11. Zoom range: 10,5-147 mm.
- CCD camera: Hamamatsu Photonics. Model: ORCA-II C-4742-98 series.
- Filters: Omega Optical. Type: Neutral Density. Flat response range [400-700] nm.
- Image acquisition and processing software: ImagePro Plus 4.1 by Media Cybernetics ®.
- White Lambertian target coating: Amercoat 741. Area: 12 m $\times$ 12 m.
- Meteorological station.

The simulation of the optical system under study is carried out with the Fiat_Lux software package (Cruz et al., 2017; Monterreal, 1999), which runs in the MATLAB environment (Higham and Higham, 2016). Once configured to use the parameterized model appropriate to the type of device, e.g., that of spherical facets when so is the studied system, Fiat_Lux allows the creation of synthetic images to compare them with reality.

As introduced, the chosen facets belong to heliostats in the CESA-I field of the PSA. The mirrors have a spherical surface and have been manufactured by the Guardian company using a second-surface glass of 4 mm width with low contents of iron. Their size is approximately 3000 mm x 1000 mm, as detailed in the next section, and they are stuck to a steel stretcher with moorings provided with silicone. The reason for choosing facets with different focal lengths as their main differentiating factor is to check if the curvature degree could affect the quality of the results. Thus, three ranges of focal length were selected, i.e., medium, long, and very long. Figure 2 shows the three facets mentioned. It includes their physical appearance, their real images acquired, their synthetic equivalents generated by Fiat_Lux according to the technical specifications provided by their manufacturer, and their comparison. Notice the difference between the iso-intensity lines of the real and the synthetic images. It supports the fact that the analyzed facets significantly deviate from their specifications.

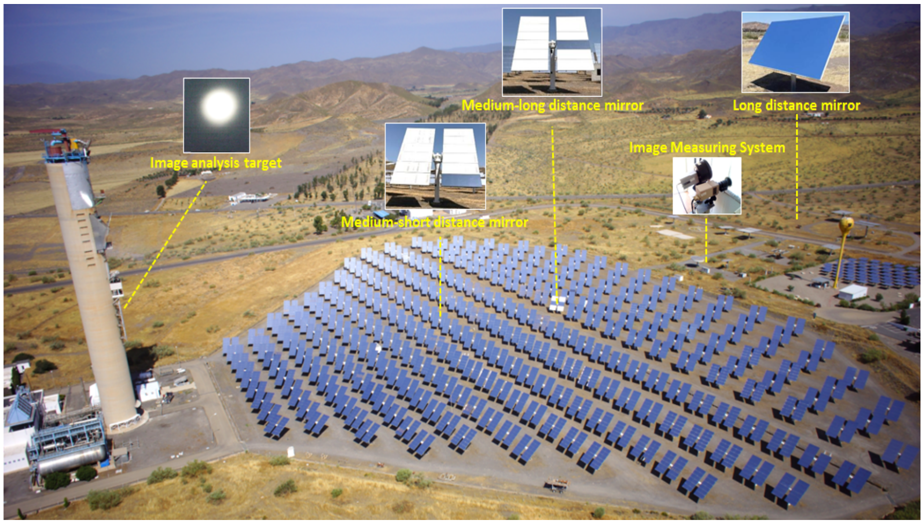


Figure 1: Image acquisition setup at the Plataforma Solar de Almería. From left to right: Image analysis target (where the real projected shapes are measured), Medium-short and Medium-long distance mirror (where the F62 and F42 facets are mounted, respectively), Image measuring system (image acquisition equipment), and Long distance mirror (where the LDM facet is mounted).

Table 1: Nominal specifications of the facets under characterization.

Facet	Surface	Dimensions (W x H) (mm)	Focal length (m)	X (m)	Y (m)	Z (m)
H0803_F62	Spherical	2995 x 1102	175	-15.144	135.200	0.893
H1100_F42	Spherical	2995 x 1102	220	-0.158	192.200	1.570
PSA_LDM	Spherical	2800 x 1605	420	-19.000	440.000	5.000

2.2. Optical model of the facets

As previously mentioned, the optical simulation tool needs to be configured to use the geometric and optical error model that correspond to the facets under characterization. The geometric model is fixed by the manufacturer based on the following factors: i) type of surface, ii) curvature or focal length (F), and iii) dimensions (wide (W) and height (H)). Table 1 contains the nominal specifications of the three facets considered in the present study. It also includes the East (X), North (Y) and Zenith (Z) coordinates of the heliostat in which they are mounted, measured from the tower base of the CESA-I field.

Regarding the optical error, ε, it is modeled as the standard deviation, σ, of the Gaussian probability distribution describing the divergence of the real normal vector of the reflecting surface from the expected one. Figure 3 depicts the facet geometry and the slope error model. While the focal length is a geometric and deterministic variable, the slope error is an angular and statistical one. It is hence necessary to take into account their different nature throughout the process of OC.

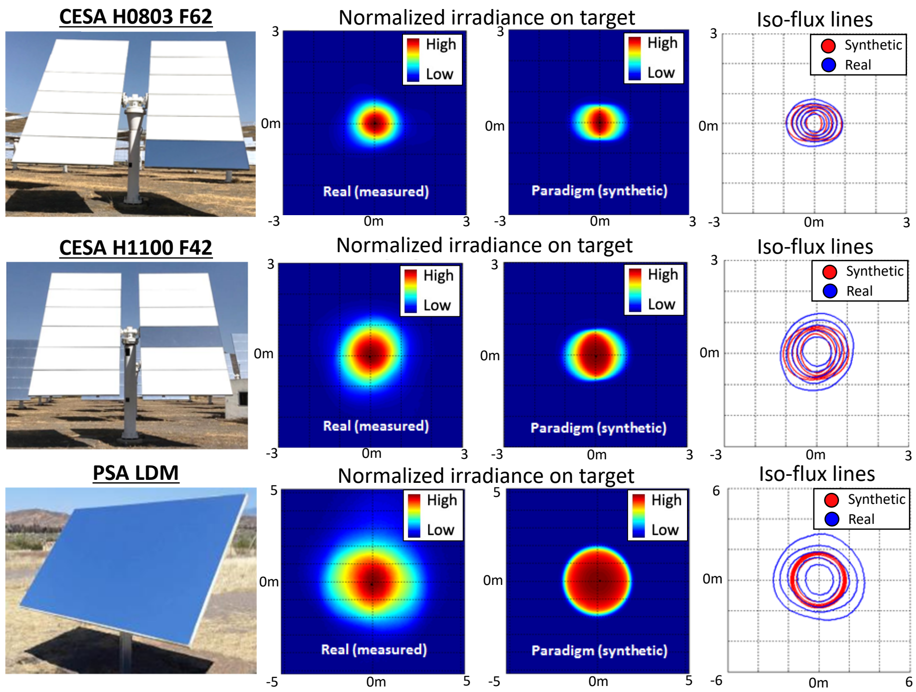


Figure 2: Real facets (left column), acquired images from measurement (central-left column), predicted images according to the manufacturers specifications (central-right column), and comparison of their iso-flux lines (right column).

2.3. Image processing and comparison

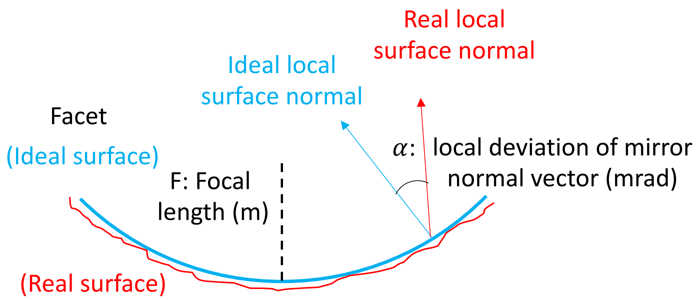
Both the traditional OC methodology and the proposed alternatives rely on comparing the real and synthetic parameter-based images. Thus, the images must be processed to make meaningful comparisons. It is also necessary to define metrics that allow assessing the similitude between real samples and synthetic proposals. All the characterization strategies share the framework defined for this purpose. It consists of the following steps described below: image normalization, segmentation and comparison.

2.3.1. Image normalization

The real and synthetic images come from different sources, i.e., ProHermes 2A and Fiat_Lux, respectively. For this reason, their gray levels (gl) have arbitrary references and cannot be directly compared. To allow their comparison, the real image and the synthetic one, which are required to be defined by matrices of the same dimension, must be modified as follows:

$$gl_{i,j}^{Norm} = \frac{gl_{i,j}}{\sum_{i,j} gl_{i,j}} \quad (1)$$

(a) Facet geometry



(b) Slope error Gaussian model

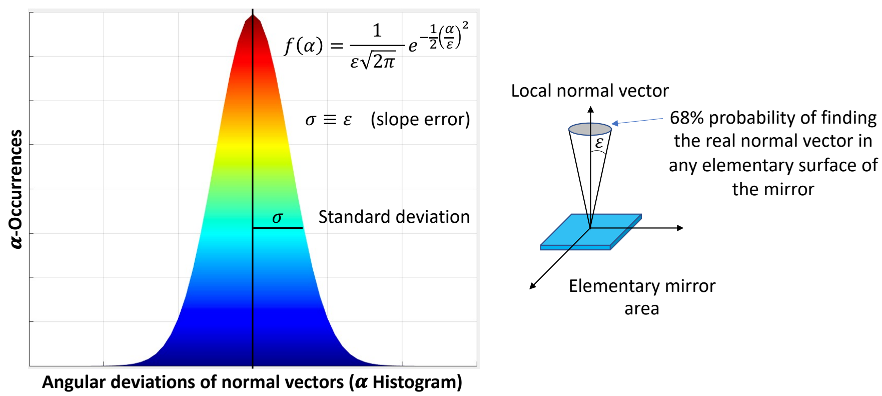


Figure 3: Facet model for characterization: a) Definition of the nominal geometry of the facet and the focal length including mirror imperfections. The latter are represented by the deviation of the normal vector from its surface at any point, also called mirror slope error. b) Population of mirror slope errors statistically described by a Gaussian distribution. Its standard deviation represents the so-called facet optical error.

where $gl_{i,j}$ refers to the gray level at pixel i, j of the image being processed, and $gl_{i,j}^{Norm}$ is the corresponding pixel after normalization. Thus, the resulting image meets the following condition:

$$\sum_{i,j} gl_{i,j}^{Norm} = 1 \tag{2}$$

In what follows, GL will designate the gray levels of the pixels of the real image (also known as the benchmark). Analogously, gl will represent those of any synthetic image, i.e., predicted by the model. This notation allows for distinguishing both types of images.

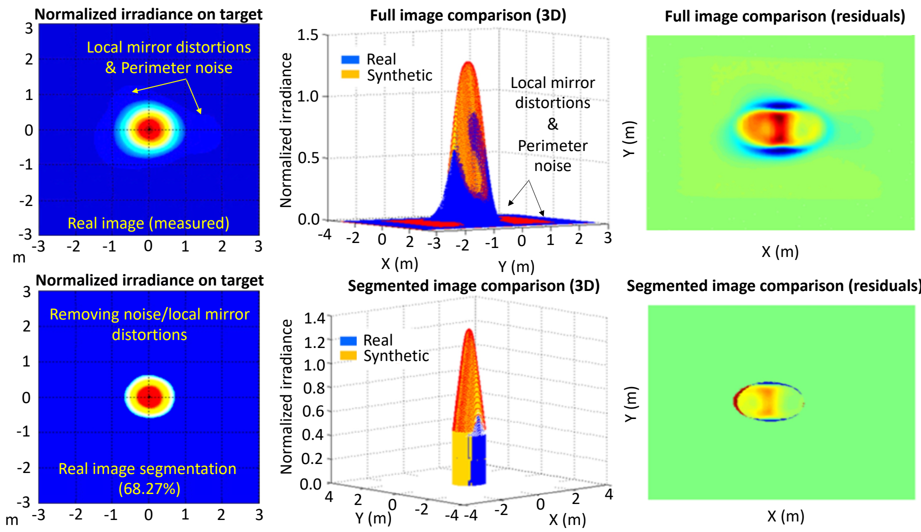


Figure 4: Understanding the comparison of sun-reflected images on target. Top-left: Occasional undesirable effects on the real image due to local distortions of the mirror and residual background noise on target. Top-central: 3D view of both real and synthetic images. Top-right: Deviation of the real image from the synthetic one by residual calculation. Down-left: removing undesirable effects on the real image. Down-central: 3D view of the real and synthetic images, free of undesirable effects. Down-right: Deviation between the real and synthetic images by means of residual calculation after removing undesirable effects.

2.3.2. Image segmentation

Real images can contain undesirable effects. For the scope of this work, the most relevant ones are the haloes caused by local mirror dislocations (or distortions) and the perimeter noise on the surface of the image analysis target. Mirror dislocations do not necessarily occur. They are very localized effects caused by the mechanical tension at the glass clamping devices that fix it to the steel frame of the facet. Their spurious nature makes them unsupported by Fiat_Lux and probably by all current simulators. The involved reflective surface is small, and so are their effects on the reflected image. These deformations are of the same order of magnitude as the noise produced by the target. They will generally appear near the image borders and feature low intensity. Regarding the perimeter noise, it is caused by the instrumentation (readout noise) and aspects such as the irregular surface of the Lambertian target. It consumes calculation time on irrelevant pixels and could misguide comparisons. Fortunately, the image segmentation process eliminates both. It considers only the part of the image that is relevant in the comparison for the OC process. Figure 4 shows the mentioned effects and how segmentation affects the image.

In terms of implementation, segmentation modifies the pixels of an input image GL^{Norm} indexed by subscripts l, m as follows:

$$\text{Segmented } GL = \begin{cases} GL_{l,m}^{Norm} = GL_{l,m}^{Norm} & \text{if } GL_{l,m}^{Norm} \geq GL_{Isoline} \\ GL_{l,m}^{Norm} = 0 & \text{Otherwise} \end{cases} \tag{3}$$

where $GL_{Isoline}$ is the threshold gray level required so that $\sum_{l,m} GL_{l,m}^{Norm} = P$, i.e., the gray level from what the summation of pixels with values equal or above is equal to the given P. Accordingly, only the pixels whose value is equal or greater than $GL_{Isoline}$ remain unaltered, while the others are set to 0. The isoline is generally chosen so that $P = 0.6827$ (68.27%). Thus, the summation of the pixels selected (taken or kept) will represent 68.27% of the intensity (accumulated gray level). The complete summation (no segmentation) would be 1, i.e., 100%. The value of P has been empirically adjusted in relation to the probability of pixels being selected according to their gray level. It results in an adequate segmentation and avoids undesirable effects (local mirror distortion and background noise) without relevant information loss for the latter comparison.

2.3.3. Image comparison

As introduced, provided that the surface type and size are known, the spherical facets under study are characterized by their slope error, ε, and focal length, F. Therefore, every set of parameters ε, F allows generating a synthetic image with the optical model to compare to reality and becomes a possible result of OC. The characterization quality increases as the difference between reality (benchmark) and the synthetic prediction decreases. This value is computed as the Root Mean Square Error (RMSE) between the benchmark and the synthetic image that results from the parameter set considered. It is formulated as follows:

$$RMSE_{segm} = \sqrt{\frac{\sum_{l,m} \left(gl_{l,m} - GL_{l,m}\right)^2}{N}} \tag{4}$$

where N is the number of pixels selected at segmentation. It is also possible to normalize the RMSE (nRMSE) by multiplying it by $100/M$, where M is the average value of the reference or benchmark image, i.e., $\overline{GL}$. This transformation allows working with values bigger than those of the plain RMSE, which are easier to remember and plot (e.g., 45.7 in nRMSE in contrast to 0.0002597 in RMSE), without altering the meaning of the results.

2.4. Optical characterization methods

Based on the previous sections, it is possible to define OC as the search for the optical parameters that minimize the difference between their corresponding synthetic image and the real or benchmark one. Hence, in practical terms, it can be addressed as an optimization problem in which the objective function to minimize is Eq. (4), and the variables are the optical parameters defining different synthetic images to compare. This section explains the alternatives considered for this process, starting from the traditional method of the PSA to the optimization algorithms studied to replace it, i.e., TLBO, UEGO, and MSIP. All of them share the operational context previously described and depicted in Figure 5. The only change is how they generate the candidate parametric combinations. The section ends with an overview of the most relevant properties of each method for their use. Thus, the reader not interested in their internals can directly go to that part.

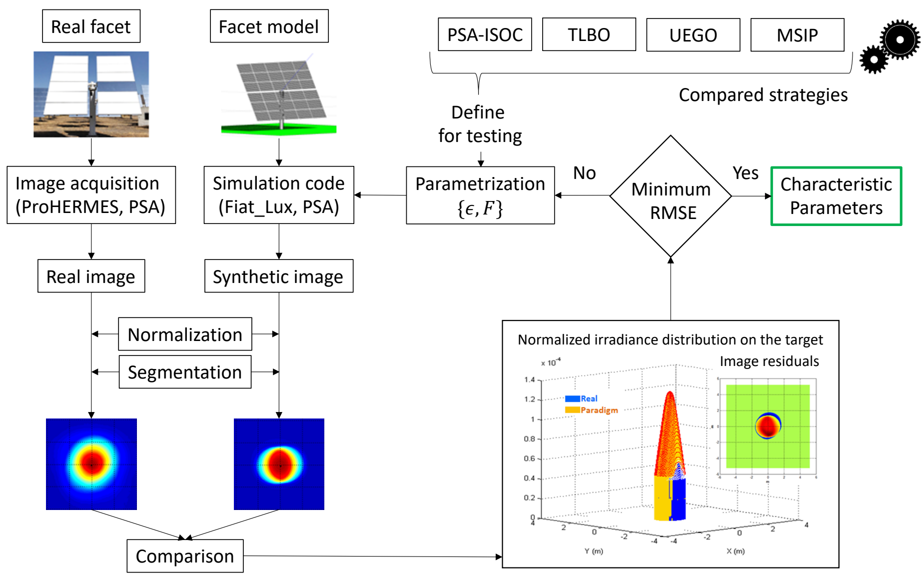


Figure 5: Common context for optical characterization. Top-left: real facet to register its image. It is next to the equivalent model to generate the synthetic image to compare after processing (bottom-left). Top-right: The OC strategies generate and study different parameter sets for the model to produce a synthetic image with the aim of finding the parameter set resulting in the minimum RMSE after comparison (bottom- right).

2.4.1. PSA Iterative Search for Optical Characterization (PSA-ISOC)

The methodology traditionally followed at the PSA, known as PSA Iterative Search for Optical Characterization (PSA-ISOC), has been improved throughout the years. Currently, the approach combines the experience of operators with the toolboxes and scripting capabilities provided by the MATLAB environment.

The characterization process starts by regularly sampling the dimensions involved, i.e., variables F and ε for the studied facet, to form a grid. The nominal value of F remains in the center of its range. The limits are generally defined in terms of percentages below and above the nominal value. For instance, 50% below and above 220 m results in a search space from 110 to 330 m. The step size is also a user-defined user parameter. It is defined considering that slight variations might not be measurable. Thus, the initial step size might be 10 or 20 m. Regarding ε, if it is known to be near a particular value, the dimension is defined the same. Otherwise, the range covers from 0 (no slope error) to a user-defined upper bound. The step size is another variable to set, and the value has to be of few milliradians in this case. The sampling technique can be generalized to all the variables involved.

After the previous definitions, the characterization process fixes the slope error to zero and explores all the sampled values for F. The method keeps a record with the best choice. According to Figure 5, assessing every combination implies generating the corresponding synthetic image, pre-processing, and comparing it to the real one to compute the RMSE. After that, F is fixed, and the exploration is repeated while focusing on the slope error this time. Finding the most appropriate slope error after having set the focal length ends the first iteration.

Once the first iteration ends, the user might decide to execute a second one. For this purpose, the ranges are centered around the previous result. The step sizes must be reduced to provide the search with more resolution. It is also possible to reduce the percentages that define the upper and lower bounds to reduce the search space. The user can execute as many iterations as desired according to the evolution of the results. However, it is not usual to perform more than three or four complete iterations. It is also relevant to highlight that the user might decide to fix the focal length variables before those linked to the slope errors, i.e., to invert the order of some iterations. It depends on the consideration of the results achieved, and it is one of the main reasons for this process to be demanding and uncertain.

2.4.2. Teaching-Learning-based Optimization (TLBO)

Teaching-Learning-based Optimization (TLBO) is a numerical optimization algorithm proposed by Rao et al. (2012). It belongs to the group of meta-heuristics (Boussaïd et al., 2013; Lindfield and Penny, 2017; Salhi, 2017), which are problem-independent optimizers. They cannot guarantee optimal solutions yet are known to achieve acceptable ones by relying on general principles. More accurately, TLBO is a population-based meta-heuristic into the Swarm Intelligence subgroup because it works with a population of candidate solutions and simulates their social interaction. Namely, it treats each solution as a student that learns from the rest and becomes a better option. This algorithm has become very popular due to its simplicity of implementation and configuration (Rao, 2016).

The configuration of TLBO consists of two parameters: the population size and the number of iterations. The optimizer starts by randomly generating as many candidate solutions as defined by the population size. Each one is a vector that contains a valid value for every variable under optimization. It is also necessary to compute the RMSE for all of them according to the procedure depicted in Figure 5. This is the value of the cost function in optimization terms. The range of each variable and the evaluation function must be provided by the user as part of the context information, but this occurs with all the options taken into account. In general, the range of variables should be the same as the widest ones considered with the traditional PSA-ISOC.

After the previous initialization stage, TLBO executes as many iterations as required. Each one consists of two consecutive steps: the teacher and the learner stages. The former simulates how students learn from their teachers and improve their skills. It tries to shift all the solutions in the population towards the best one, i.e., that with the lowest RMSE, which becomes the teacher, T. Namely, after identifying the best solution, TLBO computes a vector M in which the i component is the average of variable i in the population. Next, the optimizer applies Eq. (5) to create a modified version, S', from every candidate solution, S. This equation aims at vectors, so i refers to the i component of candidate solutions. r_i is a random real number in the range [0, 1] for each component. T_F, known as 'teaching factor', is a random integer affecting the potential amplitude of movement. Rao et al. (2012) empirically defined it to be either 1 or 2 when designing the method. Notice that every candidate solution at a particular stage and iteration shares the same random factors computed at run time. The modified solutions that outperform their original ones will replace them, while the rest are discarded.

$$S'_i = S_i + r_i\,(T_i - T_F M_i) \tag{5}$$

Regarding the learner step, it models the interaction between students. For this purpose, TLBO pairs every candidate solution with another one. Then, it applies Eq. (6) to create a modified version, S', of every existing candidate solution, S, considering the effect of its pair, W_S. The equation aims at vectors like Eq. (5). Thus, r_i refers to a real random number in the range [0, 1] and linked to the i component. The set of random factors is recomputed and remains the same for all the interactions at the present stage and iteration. For every pair, the learning stage tries to shift the worst solution towards the best. This step concludes with the replacement of the current solutions that are outperformed by their modified versions. The rest do not change.

$$S'_i = \begin{cases} S_i + r_i\,(S_i - W_i) & \text{if } error(S) < error(W) \\ S_i + r_i\,(W_i - S_i) & \text{otherwise} \end{cases} \tag{6}$$

After the method has run the requested number of cycles, it returns the best candidate solution in the population as the final result. The interested reader can find in (Rao et al., 2012) a numerical example of this optimizer in action for a better understanding.

2.4.3. Universal Evolutionary Global Optimizer (UEGO)

The Universal Evolutionary Global Optimizer, known by its acronym UEGO, is a meta-heuristic global optimizer proposed by Jelasity et al. (2001); Ortigosa et al. (2001). Like TLBO, UEGO is a population-based algorithm, but it belongs to the branch of Evolutionary Computation (Boussaïd et al., 2013). Accordingly, the algorithm manages different solutions concurrently and treats them as individuals subject to Darwinian evolution. The solutions improve their quality as

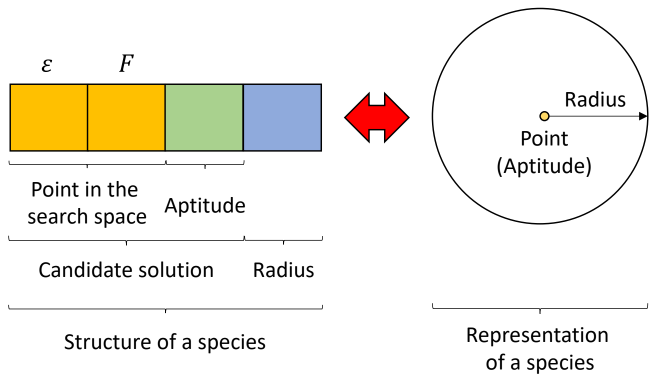


Figure 6: Depiction of species for UEGO and the target problem. In implementation terms (left), each species is an array consisting of the decision variables (ϵ, F), the value or aptitude of that parameter set as a solution (RMSE, i.e., the lower, the better), and the radius of that point as a species (Euclidean distance from the decision variables to consider nearby points equivalent). Conceptually (right), a species is a window in the search space defined by its center (the duple of (ϵ, F) for the species), and the referred distance. There are two decision variables, so the search space is bi-dimensional, and species are represented by circumferences.

they evolve. This method can be further classified as memetic (Molina et al., 2011). Thus, it adds a local search component to the simulated biological context so that solutions can improve autonomously. At the same time, UEGO is a multi-modal optimizer, which means that it seeks different optima in the search space. This algorithm has been successfully applied to many problems, such as protein folding (García-Martínez et al., 2015) and heliostat field design (Cruz et al., 2018).

The fundamental working unit of this algorithm is the species, which combines a candidate solution with an attraction radius around it. Provided a distance metric, such as the Euclidean distance for continuous variables, species are like 'windows' in the search space that focus the optimization process on different regions. These species can be created, eliminated, moved, and merged throughout the operation of UEGO. In fact, in practical terms, the optimizer can be thought of as a method for managing a list, i.e., a species population. Figure 6 shows the structure and meaning of a species considering the slope error (ε) and the focal distance (F) of a hypothetical facet under OC as the optimization variables. The slope error and the focal distance defining the central point of the species also form a solution to the problem. Hence, its value as a solution, known as aptitude in Evolutionary Computation, is also registered. As it refers to the RMSE obtained after simulating the corresponding parameter set, the lower this value is, the better solution this configuration represents.

Aside from the problem context, i.e., the variable bounds and the objective function, UEGO takes the following parameters: i) the maximum number of species, ii) the maximum number of evaluations of the objective function, iii) the minimum radius between species, and iv) the number of iterations (search levels in UEGO). Nevertheless, take into account that the selected local search algorithm might require extra parameters.

The algorithm starts by creating an initial species. Its center is a random feasible point, and its radius is set to the diameter of the search space to cover it completely. Then, UEGO launches the selected local optimizer from the center of this species. This point will be updated as the method finds better ones, which is equivalent to moving the species. These steps define the first level of search.

After the initialization, UEGO executes a loop with the remaining levels of search. Each one consists of the following steps. Firstly, UEGO computes the radius to be assigned to any new species created at that level. Radii decrease with the search level according to a geometrical progression, and the last one is the user-given input. This strategy corresponds to a cooling component that progressively reduces mobility to promote convergence. UEGO also computes the budget of objective function calls to create and locally optimize species. The former is three times the maximum number of species, and the latter increases with the level, as radii decrease. However, UEGO has mechanisms to save function evaluations, such as removing redundant species, so it might not consume all of them.

Secondly, UEGO creates as many random species in the regions of the existing ones as allowed by the budget. Then, the algorithm fuses any species whose centers are nearer than the radius assigned to the current level. When fusing two species, the resulting one keeps the longest radius, which tries to avoid premature convergence, and takes the better candidate solution as its new center. Next, if the population is larger than allowed, the excess is removed, starting with the species having the shortest radius. After that, UEGO launches the local optimization component from each species and considering the function evaluation budget. To conclude every search level, as the local optimizer moves the species centers to better candidate solutions independently, UEGO repeats the fusing process.

With respect to the local search component, the Solis and Wet's algorithm has been selected (Molina et al., 2011; Solis and Wets, 1981). This method is a stochastic hill-climber that starts at a given point, i.e., the center of a species when coupled to UEGO, and takes improving steps in random directions. The amplitude of jumps is scaled with the number of consecutive successful (improving) and discarded (non-improving) movements. This local search algorithm has been selected because it has no specific requirements from the objective function, and it has already been successfully coupled with UEGO (Cruz et al., 2018; Jelasity et al., 2001).

After having executed as many search levels as requested, the output of UEGO is the final list of species. According to the multi-modal approach of the algorithm and provided that it has converged, the surviving species are expected to be different optimal solutions.

2.4.4. Multi-Start Interior-Point (MSIP)

The optimizer referred to as Multi-Start Interior-Point, abbreviated as MSIP, results from coupling an Interior-Point algorithm (Griva et al., 2009) with a stochastic Multi-Start technique (Redondo et al., 2013; Salhi, 2017).

Interior-Point methods form a group of algorithms for addressing linear and non-linear optimization problems (Griva et al., 2009). They are characterized by keeping the exploration in the feasible region of the search space employing different methods. The Interior-Point algorithm considered in this work belongs to the FMinCon (FMC) solver included in the Optimization Toolbox of MATLAB (Branch and Grace, 2020; López, 2014). The method addresses the target problem by solving a sequence of approximations that result from adding slack variables and a barrier function (Byrd et al., 2000). The resulting instances are simpler to solve than the original one.

FMC takes as input an initial point to start the search, and it might impact the final result depending on the problem type. At each iteration, it can choose one of two alternatives to solve the approximate problem. The first option, which is also the preferred one, is to take a direct or Newton step that applies a linear approximation. The second one is to take a conjugate gradient step with a trust region. This latter option is only selected when it is not possible to apply the previous one, for example, because the approximate problem is not locally convex near the current position.

The referred Interior-Point algorithm will only find the optimal solution for convex problems. Otherwise, the result might be a local optimum. The convexity of the target problem has not been certified. Moreover, according to preliminary experimentation, the solutions found by FMC vary with the starting points. Thus, the multi-start component is in charge of randomly generating different initial points and launching FMC independently from each one. It controls the total number of function evaluations and records the best result achieved so far, which ultimately becomes the problem solution. This approach serves to escape from local optima by focusing the seek on different regions of the search space.

2.4.5. Overview of the OC methods

After explaining the four options considered for finding the optical parameters of the studied facets, Table 2 provides the reader with a summary of the main properties of each. The first column contains the most relevant aspects to take into consideration. After that, there is a column per method with the corresponding details. PSA-ISOC stands out as the only deterministic method, i.e., it always returns the same result (as long as the same expert decisions are taken). However, it is also the most tedious and difficult to apply due to its inherent link to an expert. The others are solvers that only need to be appropriately configured and launched. Nevertheless, their stochastic nature can make their output vary among executions, so several runs might be needed. Among them, UEGO seems the hardest to configure after a first glance, but configuring the FMC part of MSIP can be challenging, as it has more than ten parameters according to the official documentation. Hence, TLBO is the best balanced in this concern as it only takes two parameters. Finally, notice that PSA-ISOC cannot be independently applied without an expert and time, which is one of the problems that this work aims to correct, and MSIP requires a software license

Table 2: Main properties of the different methods considered for OC.

	PSA-ISOC	TLBO	UEGO	MSIP
Type	Semi-automatic grid search	Meta-heuristic (Swarm Intelligence)	Meta-heuristic (Evolutionary Computation)	Interior-Point in Multi-Start component
Output stability	Deterministic (Taking the same decisions)	Stohastic	Stochastic	Stochastic
Human interaction	High (Interactive stages)	Minimal (Setup)	Minimal (Setup)	Minimal (Setup)
Setup complexity	High (Under continuous adaption)	Low (2 parameters)	Medium (4 parameters)	Minimal (1 parameter, using the defaults of FMinCon)
Availability	Not applicable (Expert-dependent)	Open-source implementation at Cruz (2021a)	Open-source implementation at Cruz (2021b)	Optimization Toolbox required

for FMC. On the contrary, we have open-source implementations of TLBO and UEGO, simplifying their use (or even modification).

3. Experimentation and results

3.1. Environment and configuration

The present study has considered three spherical heliostat facets to characterize: F62, F42, and LDM. After recording the benchmark image of each facet as described in Section 1 and Section 2, the different characterization processes have been carried out in a computational environment. At that point, the tools used are MATLAB 2018b, the auxiliary routines for image management, the Fiat_Lux optical model from the PSA, and the optimizers involved.

There are four alternative OC strategies, the traditional and semi-automatic PSA-ISOC method, and the optimizers TLBO, UEGO, and MSIP, which explore the parameter search space autonomously after being configured. The first one has been executed, as usual, by one of the experts in charge of OC at the PSA. The others have been launched independently at the University of Almería (UAL). The MATLAB implementations of TLBO and UEGO have been developed at the UAL. The same occurs with the multi-start component that manages the FMinCon software package by MathWorks used to define MSIP. The computer used at the PSA features an Intel Core i5-8265U 4-core processor and 16 GB of RAM. The one used at the UAL has an Intel i7-4790 4-core processor and 32 GB of RAM.

The PSA-ISOC procedure was applied at the PSA, as usual, to minimize the same objective or cost function as the optimizers. It covered the following aspects:

1. Facet geometry: Spherical

(a) Variable linked to the search: Focal distance (m).
(b) Nature of the variable: Deterministic.
(c) Starting Value (SV) (manufacturer's data):
   i. 220.0 m (F42)
   ii. 175.0 m (F62)
   iii. 420.0 (LDM)
(d) Search range:
   i. $[F_{min} = 90.0, F_{max} = 330.0]$ (F42, F62)
   ii. $[F_{min} = 150.0, F_{max} = 650.0]$ (LDM)

2. Mirror surface state.
   (a) Variable linked to the search: slope error (mrad)
   (b) Nature of the variable: Statistical.
   (c) SV: 0.0 mrad (no manufacturer's data).
   (d) Search range:
      i. $[\epsilon_{min} = 0.0, \epsilon_{max} = 3.0]$ (F42, F62)
      ii. $[\epsilon_{min} = 0.0, \epsilon_{max} = 2.5]$ (LDM)

Two variants of PSA-ISOC were tested for each facet:

Direct: It starts at the initial configuration. It assumes that the SV of the slope error is correct, so the search starts by testing the focal length of the facet, which is the variable of deterministic nature.

Inverse: It starts at the initial configuration. It assumes that the focal F of the facet is correct and its SV is the nominal one given by the manufacturer, so the search starts by testing the slope error, which is the variable of statistical nature.

In both, the best value found for the variable considered is recorded and used when switching to the other dimension, and the process is repeated until finding the lowest error. Each variant generates its own final result, and the one ultimately selected is that producing the smallest error.

The optimizers inherit the initial ranges of each variable and facet described above. TLBO has been configured to work with a population of 12 candidate solutions for 30 cycles in all the cases. This configuration approximately results in 730 objective function evaluations. It implies the same number of simulations with Fiat_Lux generating a synthetic image to compare to the benchmark. This value is based on the number of evaluations that the simplest PSA-ISOC would execute, according to the expert. MSIP and UEGO have been configured to approximately make the same number of function evaluations. More specifically, UEGO has been configured to consume up to 1400 function evaluations, maintain up to 12 individuals, run 16 levels of search, and consider a minimum radius of 4.0 in the search space. Since it includes strategies to save function evaluations after convergence, the number stays in the desired range. Finally, MSIP has been directly provided with the reference number of function evaluations.

Regarding the computational effort, the methods considered can be divided into two groups. Let the traditional expert-based PSA-ISOC be on one side and the standard optimization algorithms UEGO, TLBO, and MSIP on the other. This division is equivalent to the degree of human interaction detailed in Table 2. It can be reformulated as the methods of high human interaction (PSA-ISOC) compared to those mainly automatic (only requiring to be configured once), i.e., UEGO, TLBO, and MSIP. In general, the time required by the traditional expert-based PSA-ISOC will be significantly higher than the rest due to the necessity of supervising its progress and making decisions. Hence, PSA-ISOC could only compete with the standard optimization algorithms in terms of quality. More specifically, the expert invested 2 days of work applying the PSA-ISCO to every facet, i.e., 16 hours for each case. On the contrary, the time of any optimizer mainly depends on the number of function evaluations, which makes them equivalent in terms of computational effort. The reason is that the objective function is computationally demanding (e.g., its evaluation takes approximately 45 seconds for F42), which makes the cost of the internal steps of any optimizer negligible. As all the standard optimizers have been configured to launch approximately 730 evaluations, their running time is virtually the same. Namely, all the optimizers took approximately 9.5 hours for F42, 8.5 hours for F62, and 4.5 hours for LDM in the non-dedicated UAL computer. Thus, the execution time is not relevant for choosing one of the optimizers.

3.2. Problem considerations

In contrast to a classic optimization problem analytically addressed, e.g., finding the minimum of the parabola $f(x) = x^2$, we face two levels of stochasticity. The first one is linked to the objective function evaluation, which involves the simulation an generation of synthetic images to compare. More specifically, if one thinks of evaluating the referred parabola, the values computed are deterministic and do not vary. For instance, $f(5) = 5^2 = 25$, and the result does not change. On the contrary, the results of generating and comparing synthetic images to the real one vary. The reason is the slope error, modeled by a Gaussian distribution. This aspect affects the creation of synthetic images by Fiat_Lux with random yet normally distributed values. In this context, given two similar parameter sets, it is possible that one evaluation using Fiat_Lux returns than the first one is slightly better, while the next one results in the opposite interpretation. It is like looking for the minimum point of a function that moves slightly. We face this problem by making the objective function (its internal simulations) deterministic artificially. Namely, we fix the same seed at evaluation, i.e., the one used by Fiat_Lux, for a complete execution of any optimizer. Roughly speaking, the seed in Computer Science is the input of random number generation algorithms defining an infinite sequence of pseudo-random numbers (Matsumoto and Nishimura, 1998). For the same seed, they produce the same numbers. Hence, this is like taking a snapshot of the moving surface of the objective function or freezing it.

The second level of stochasticity is linked to the optimization methods considered. Even with a fixed surface to explore, they also use random numbers, i.e., they are stochastic. This property means that a certain optimizer might return a different result for the same problem and configuration. In other words, its results are not deterministic either (also in contrast to the analytical approach of the example above). This behavior is coupled with the variation of the seed defining the objective function. Regardless, the stochastic component of the Fiat_Lux simulations is enough to impact the whole OC process. Thus, it is necessary to handle this situation.

For this purpose, every optimizer has been independently executed with five different fixed seeds for the objective function (Fiat_Lux simulation). Accordingly, for each facet and method, we ultimately have 5 equivalent parameter sets. We cannot say that one is better than the rest because they were obtained from a different snapshot of the objective function. Thus, the final parameter set from the particular optimizer an facet is the average of their independent results. For example, assuming a particular facet and two simulation seeds varying the shape of the objective function, we could have registered from one of the optimizers the following results: (0.52 mrad, 175 m) and (0.58 mrad, 185 m). Then, its result for the studied facet would be (0.55 mrad, 180 m), i.e., the average of the variables.

However, that parameter set might perform bad in reality, especially if the optimizer mis-converged. They cannot even be certainly ranked: each parameter set has its own value for its frozen function, but the quality of the average point cannot be computed in that way. Thus, each parameter set is ultimately evaluated an compared by studying it under 100 different seeds for the simulation, i.e., by evaluating with 100 different snapshots of the same moving objective function (which cannot be fixed due to its statistical definition). If the parameter sets adapt well to the different variations of the function, they are robust enough. In other words, the robustness of every result must be assessed by considering multiple evaluations (synthetic image generation and comparison) with different random numbers. For this reason, it is always advisable to vary them during the seek, even if the OC methods are deterministic.

It is interesting to end this subsection with a more detailed explanation of how Fiat_Lux works in practical terms. It receives as input a picture of the sun, and it is taken with the same digital camera used for the images of the facets over the target. It is an angularly-calibrated image, which means that we know the mrad subtended for any pixel in the picture of the sun from the camera. This can be applied to any element in the concentrator too, given the proximity between both. In this regard, Fiat_Lux differs from standard Monte-Carlo Ray-Tracing (MCRT), which associates the uncertainty of the optical system to the direction of the incident ray. For it, the uncertainty at simulation is caused by the deviation in the orientation of the vector perpendicular to every element of the surface, dS, on the mirror (slope error). This orientation differs from the nominal one and is modeled by a normal distribution statistically describing the deviation of the corresponding perpendicular vector, and ultimately affects the direction of every ray reflected at every dS. Therefore, the unavoidable uncertainty is handled by considering multiple seeds, as detailed above. Each Fiat_Lux simulation has processed approximately 930 000 000 rays.

3.3. Optical characterization results

Figure 7 depicts intermediate results of the PSA-ISOC process for F42. It allows visualizing the impact on the successive approximations of the horizontal and vertical normalized intensity profiles of the real and synthetic images. Figure 8 displays the slope error and focal length through the stages of the Direct and Inverse variants of PSA-ISOC for the same F42 facet. The asymptotic evolution of the variables shows the proximity of the minimum error. The behavior is similar for F62 and LDM, so they are omitted due to space limitations. Regardless, the PSA-ISOC method has been outperformed by optimization for every facet.

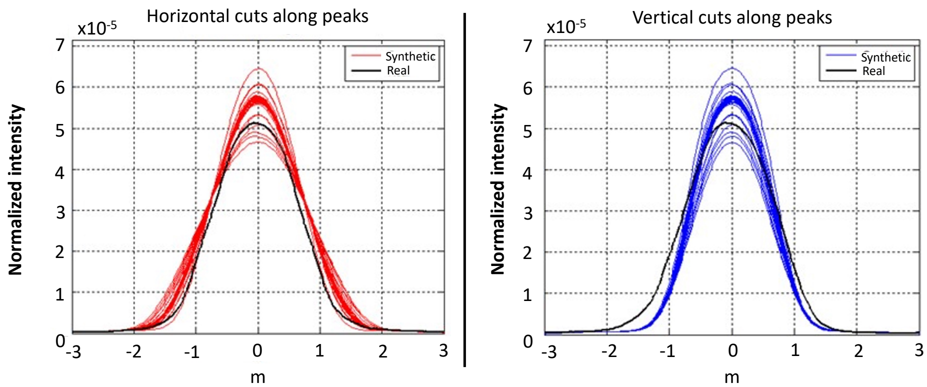


Figure 7: Horizontal (left) and vertical (right) slices across the centroid of the normalized intensity of the real image (black) acquired from the F42 facet. The red and blue slices correspond to synthetic images generated with different model parameters considered by the PSA-ISOC method.

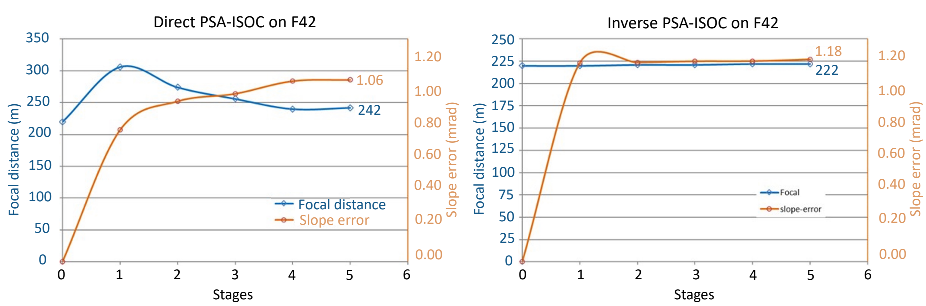


Figure 8: Partial results of PSA-ISOC for F42. Left (Direct PSA-ISOC): The blue line represents the evolution of the searched focal length of the F42 facet from its initial (nominal) value to the best value minimizing the cost function. The brown line shows this evolution for the slope error. Right: It contains the same information for the Inverse PSA-ISOC variant.

Regarding the results of advanced optimization, Table 3 contains the nominal values of the parameters defining each facet, i.e., slope error and focal distance, alongside the corresponding values achieved through characterization. It also

		Nominal optical parameters		Best optical parameters				
Facet	Geometry	Slope error (mrad)	Focal length (m)	Slope error (mrad)	Focal length (m)	Method	nRMSE	Confidence Interval (90%)
F62	Spherical	Unknown	175	0.76	171	UEGO	89.46	18.1%
F42	Spherical	Unknown	220	1.18	211	TLBO	34.20	10.7%
LDM	Spherical	Unknown	420	0.81	185	TLBO	17.98	6.7%

Table 3: Final results of the optical characterization process.

includes the method that found the best solution, represented by its nRMSE and 90% confidence interval. Unknown values refer to the fact that manufacturers did not provide information about this parameter for the facets built.

Figure 9 shows the robustness analysis of the final result of each method and facet. Each solution is represented by its average nRMSE registered after 100 independent evaluations with different seeds. The figure also displays the corresponding 90% confidence interval of each sample. As previously explained, any of the 100 nRMSE instances could be the most realistic one. Thus, they are replaced with two representative statistics, i.e., the average and the standard deviation, as the most probable value and the observed scattering, respectively.

Figure 9 confirms the viability of replacing the traditional characterization approach with optimization algorithms, which was a fundamental goal in the present study. More specifically:

- The nRMSE of PSA-ISOC is similar to those of the best-performing numerical optimizers, i.e., TLBO and UEGO. More specifically, one can compare the nRMSE obtained in the test of 100 independent evaluations for each result found by PSA-ISOC to that of the best-performing optimizer, i.e., lowest average nRMSE. For F62, the result of UEGO deviates 0.55% up from that of PSA-ISOC. For F42, the result of TLBO deviates up 5.32%. Finally, for LDM, TLBO deviates up 1.87%.

- The solutions of PSA-ISOC fall into the confidence interval defined by the best-performing numerical optimizers.

- For all the facets, the lowest average nRMSE comes from a numerical optimizer. Hence, they could replace the traditional PSA-ISOC while providing similar quality and requiring less supervision by the experts.

Figure 10 extends Figure 2 by comparing the iso-intensity lines of the real images of the facets to their synthetic equivalents according to the best parameters found after characterization. As can be seen, the iso-intensity lines of the real images are more similar to those of the synthetic ones generated with the characterized configuration than to those using the nominal values.

The similarity between the actual images of the facets and their synthetic equivalent can be further studied with statistical tools. In this context, the pixel-to-pixel deviation (*px2px*) between the real image and the one simulated with the parameters obtained after characterization has been analyzed. Figure 11 shows this study (left side) and the corresponding histograms of occurrences (right side) for the three facets considered.

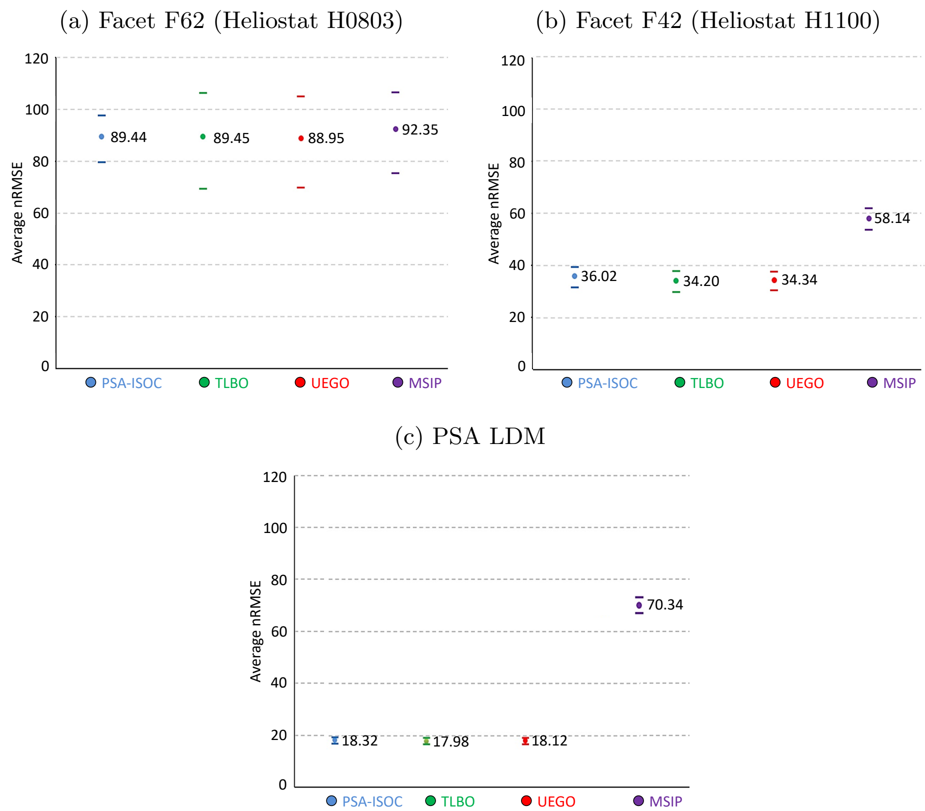


Figure 9: Quality of the results. The optical parameters found by each method for every facet have been used to compare the resulting synthetic image to its reference under 100 different random seeds to confirm their generality. The numbers correspond to the average nRMSE obtained with each parameter set after the 100 independent comparisons. The associated bars indicate the 90% confidence interval of these results.

Table 4 shows three different metrics of pixel-to-pixel deviation between the actual images and their synthetic equivalents found after characterization and their occurrence percentages. Namely, Type A refers to the most frequent deviation of the synthetic image from the actual one. Type B is the name of the highest deviation above the real image. Type C is how the highest deviations below reality are tagged.

The first group shows if the characterized image has either overestimated or underestimated the intensity in the benchmark, as well as the magnitude of the most frequent deviation. For instance, the optical parameters found for the facet F62 cause 96.75% of the pixels in the synthetic image to be 0.33% above the actual values. Accordingly, the characterization predominantly overestimates the intensity. However, the opposite situation occurs with F42 and LDM.

This analysis would be biased without considering the magnitude and fre-

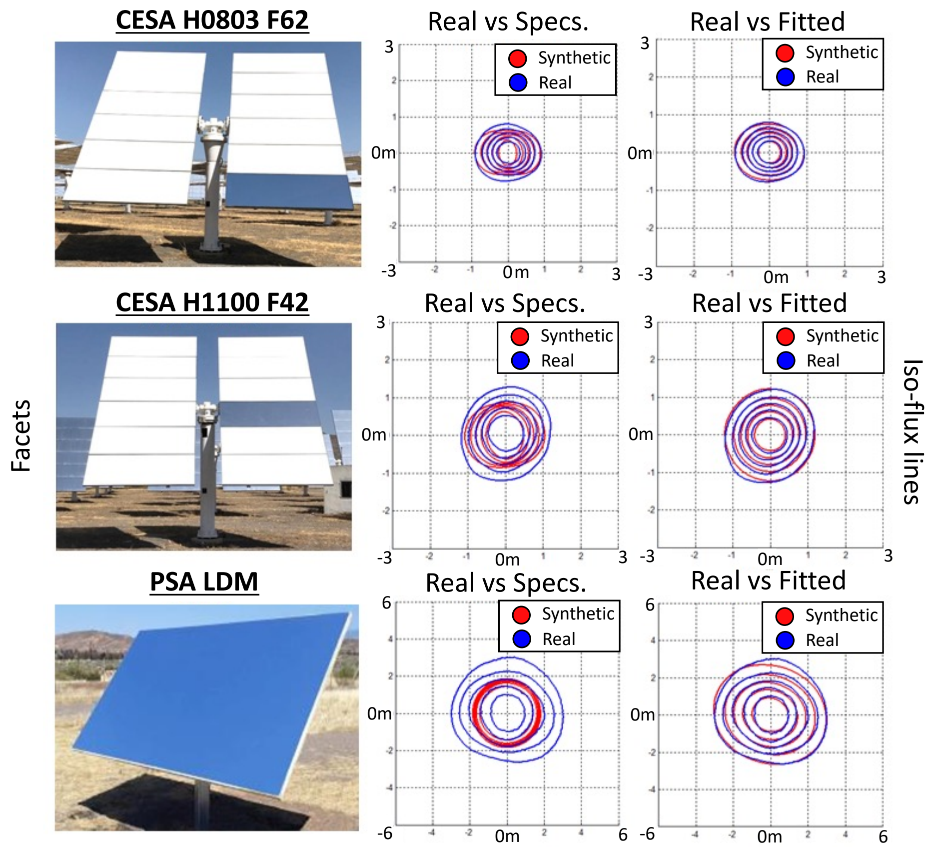


Figure 10: Iso-intensity lines of the real images of each facet (left) compared to the synthetic equivalent using: the nominal parameters (center) and those obtained through characterization (right).

Facet	Type A px2px deviation	Occurrence	Type B px2px deviation	Occurrence	Type C px2px deviation	Occurrence	Others
F62	0.33%	96.75%	13.9%	0.007%	-13.2%	0.006%	3.24%
F42	-0.32%	93.39%	20.0%	0.009%	-20.7%	0.017%	6.58%
LDM	-0.62%	88.16%	15.9%	0.001%	-14.1%	0.001%	11.86%

Table 4: Statistical analysis of the best synthetic images obtained through characterization.

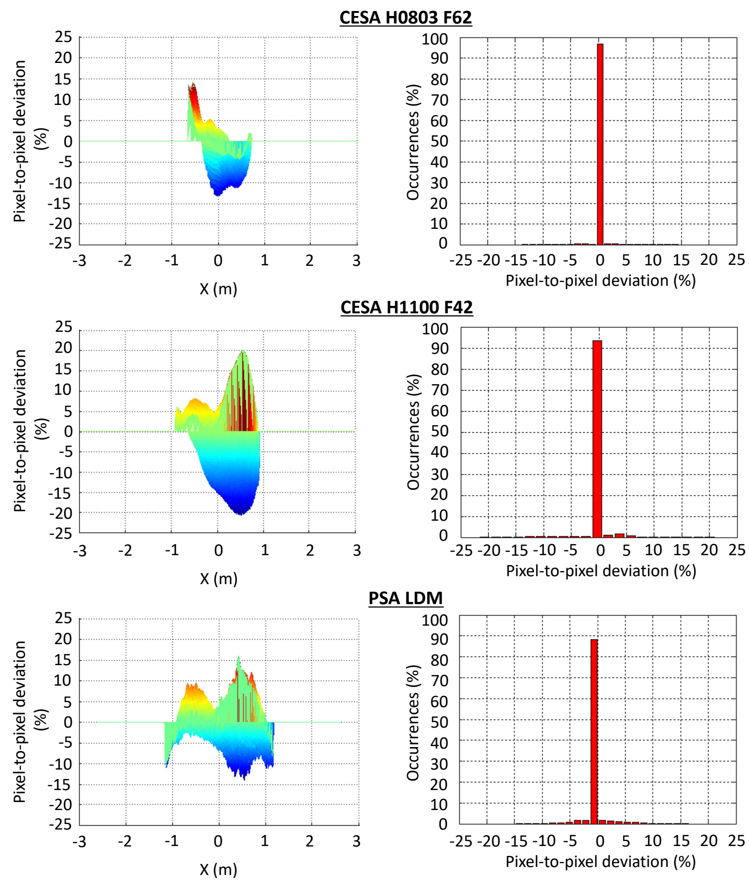


Figure 11: Pixel-to-pixel deviation between the real image and the best characterization result (left) and corresponding histogram of occurrences (right) for each facet. The best parameter set found through optimization for every facet has been used to generate the corresponding synthetic image with Fiat_Lux. These images have been compared pixel by pixel to their corresponding reference. The comparison computes the percentage deviation of each pixel in the synthetic image with respect to its equivalent in the one experimentally obtained (reference) generating a histogram of percentage deviations.

quency of the highest deviations between both sorts of images, which can be determinant to assess characterization. For example, the result of F42 shows deviations up to 20.0% above its benchmark, with an occurrence of 0.009%. For LDM, 15.9% are above its benchmark and the occurrence is 0.001%, i.e., 9 times less. Regarding the highest deviations below the benchmark, the percentages are similar (-20.7% and -14.1%, respectively). However, their occurrences are 0.017% and 0.001%, respectively, i.e., 17 times less for the facet LDM. Therefore, it can be concluded that LDM is the facet that has been characterized the best. This idea is supported by Figure 11 and the position of its results in the nRMSE ranking, in which its most pronounced deviations from the benchmark are also rare.

4. Conclusions and future work

This work has two main goals. The first one is to warn about the necessity of both an initial and a routine assessment of the optical quality of the heliostats in commercial solar power tower plants throughout their useful life. The second one is to demonstrate that this process can be carried out as long as effective alternatives to traditional indirect optical characterization, such as the method followed at the Plataforma Solar de Almería (PSA), are developed. The viability of this approach depends on minimizing two fundamental aspects of current strategies: i) the requirement of constant participation of highly qualified staff, and ii) the time linked to traditional methods, which are mainly iterative searches with notable heuristic or expert-specific components.

A common experimentation framework has been defined with the experts of the PSA. In this context, the traditional iterative optical characterization method of the PSA (PSA-ISOC) has been compared to using three existing numerical optimizers, UEGO, TLBO, and MSIP. The study has not been limited to obtaining an independent result to optical characterization instances. It has also covered how to find the optimal solution among the proposals of each method. The experimentation has considered three heliostat facets. Two of them belong to standard heliostats of the CESA-I field of the PSA, while the third one is a facet prototype featuring a long focal distance. The results obtained confirm that for simple optical systems, such as heliostat facets, the numerical optimizers TLBO and UEGO achieve the optimal solution for the optical characterization problem. More precisely, for the three facets considered, the traditional semi-automatic process of the PSA, PSA-ISOC, deviates 0.55%, 1.87%, and 5.32% in nRMSE, respectively, from the results obtained by the automated execution of the optimizers. Besides, the PSA-ISOC results fall into the confidence interval linked to those achieved by the optimizers. Therefore, these methods seem valid to approach optical characterization processes without requiring constant human interaction. The decision between them is up to the user, but TLBO is simpler to tune and implement.

For future work, the aim is to generalize the optical characterization method to cover complete facet-based heliostats. Upon success, it will be possible to include the technique as part of the routine revision and maintenance tasks of heliostats in solar power tower plants. This update allows monitoring their optical performance throughout their useful life and, thus, controlling the economic expectations linked to their real production capacity.

Acknowledgments

This work is part of the R&D project RTI2018-095993-B-I00, funded by MCIN/ AEI/10.13039/501100011033/ and ERDF ("A way to make Europe"). Part ot the research work of this article received ERDF funds from the Spanish government within the framework of the SOLTERMIN project (Ref. ENE2017-83973-R) of the *Ministerio de Economía, Industria y Competitividad* (Spanish Ministry of Economy, Industry and Competitiveness). It has also been supported by the Andalusian Regional Government through the grant: "*Proyectos de Excelencia*" (P18-RT-1193), and the University of Almería through the grant: "*Ayudas a proyectos de investigación I+D+I en el marco del Programa Operativo FEDER 2014-20* (UAL18-TIC-A020-B). N.C. Cruz is supported by the Ministry of Economic Transformation, Industry, Knowledge and Universities from the Andalusian government.

References

A. J. Gallego, E. F. Camacho, On the optimization of flux distribution with flat receivers: A distributed approach, Sol. Energy 160 (2018) 117–129.

D. Sah, K. Lodhi, C. Kant, P. Saini, S. Kumar, Structural composition and thermal stability of extracted eva from silicon solar modules waste, Sol. Energy 211 (2020) 74–81.

E. Kabir, P. Kumar, S. Kumar, A. A. Adelodun, K. H. Kim, Solar energy: Potential and future prospects, Renew. Sust. Energ. Rev. 82 (2018) 894–900.

V. S. Reddy, S. C. Kaushik, K. R. Ranjan, S. K. Tyagi, State-of-the-art of solar thermal power plantsa review, Renew. Sust. Energ. Rev. 27 (2013) 258–273.

M. Saghafifar, M. Gadalla, K. Mohammadi, Thermo-economic analysis and optimization of heliostat fields using aineh code: Analysis of implementation of non-equal heliostats (aineh), Renew. Energy 135 (2019) 920–935.

A. de la Calle, A. Bayon, J. Pye, Techno-economic assessment of a high-efficiency, low-cost solar-thermal power system with sodium receiver, phase-change material storage, and supercritical CO_2 recompression Brayton cycle, Solar Energy 199 (2020) 885–900.

N. C. Cruz, J. D. Álvarez, J. L. Redondo, M. Berenguel, R. Klempous, P. M. Ortigosa, A simple and effective heuristic control system for the heliostat field of solar power tower plants, Acta Polytech. Hungarica 17 (2020) 7–26.

J. Fernández-Reche, Reflectance measurement in solar tower heliostats fields, Sol. Energy 80 (2006) 779–786.

N. C. Cruz, J. L. Redondo, M. Berenguel, J. D. Álvarez, P. M. Ortigosa, Review of software for optical analyzing and optimizing heliostat fields, Renew. Sust. Energ. Rev. 72 (2017) 1001–1018.

L. García, M. Burisch, M. Sánchez, Spillage estimation in a heliostats field for solar field optimization, Energy Procedia 69 (2015) 1269–1276.

M. Kiera, On the interpretation of measured heliostat images, Technical Report, INTERATOM Internal Report, GAST Project, 1985.

S. A. Jones, M. E. Robert, R. M. Houser, Recent results on the optical performance of Solar Two heliostats, Sol. Eng. 1 (1995) 581–596.

J. W. Strachan, R. M. Houser, Testing and evaluation of large-area heliostats for solar thermal applications, Technical Report, Sandia National Lab.(SNL-NM), Albuquerque, NM (United States), 1993.

J. W. Strachan, Revisiting the BCS, a measurement system for characterizing the optics of solar collectors, in: 39th International Instrumentation Symposium, 1993, pp. 2–6.

F. Biggs, C. N. Vittitoe, The HELIOS model for the optical behavior of reflecting solar concentrators, Technical Report, Sandia National Laboratory, Albuquerque, NM (United States), 1979.

S. Cordes, Test and evaluation of the Schlaich, Bergermann and Partner Heliostat Prototype Concentrator, Technical Report, SolarPACES Technical Report No. III-4/94, 1994.

R. Monterreal, Development and testing of a new 100m^2 glass-metal heliostat, in: Proceedings of ASME International Solar Energy Conference, 1997, pp. 1–10.

R. Monterreal, A new computer code for solar concentrating optics simulation, J. Phys., IV 9 (1999) Pr3–77.

D. L. King, Beam quality and tracking-accuracy evaluation of second-generation and Barstow production heliostats, Technical Report, Sandia National Laboratories. Albuquerque, NM (United States), 1982.

J. A. Avellaner, Optical characterization of the facets of a heliostat, Rev. Phys. Appl. 15 (1980) 169–173.

P. Bendt, A. Rabl, Optical analysis of point focus parabolic radiation concentrators, Appl. Optics 20 (1981) 674–683.

M. Kiera, W. Schiel, Measurement and analysis of heliostat images, J. Sol. Energy Eng. 111 (1989) 2–9.

S. Kammel, Deflectometry for quality control of specular surfaces, TM-Tech. Mess. 70 (2003) 193–198.

R. Monterreal, R. Enrique, J. Fernández-Reche, An improved methodology for heliostat testing and evaluation at the plataforma solar de almería, in: AIP Conference Proceedings, volume 1850, AIP Publishing LLC, 2017, p. 030036.

K. Pottler, E. Lu¨pfert, G. H. G. Johnston, M. R. Shortis, Photogrammetry: A powerful tool for geometric analysis of solar concentrators and their components, J. Sol. Energy Eng. 127 (2005) 94–101.

P. Garcia, A. Ferriere, J. Bezian, Codes for solar flux calculation dedicated to central receiver system applications: A comparative review, Solar Energy 82 (2008) 189–197.

J. Nocedal, S. Wright, Numerical optimization, Springer Science & Business Media, 2006.

A. Törn, A. Žilinskas, Global optimization, volume 350, Springer, 1989.

R. Monterreal, N. C. Cruz, J. L. Redondo, J. Fernández-Reche, R. Enrique, P. M. Ortigosa, On the optical characterization of heliostats through computational optimization, in: AIP Conference Proceedings, volume 2445, AIP Publishing LLC, 2022, p. 070007.

S. Salhi, Heuristic search: The emerging science of problem solving, Springer, 2017.

R. V. Rao, V. J. Savsani, D. P. Vakharia, Teaching–learning-based optimization: an optimization method for continuous non-linear large scale problems, Inf. Sci. 183 (2012) 1–15.

M. Jelasity, P. M. Ortigosa, I. García, UEGO, an abstract clustering technique for multimodal global optimization, J. Heuristics 7 (2001) 215–233.

I. Griva, S. G. Nash, A. Sofer, Linear and nonlinear optimization, volume 108, Siam, 2009.

N. C. Cruz, R. Ferri-García, J. D. Álvarez, J. L. Redondo, J. Fernández-Reche, M. Berenguel, R. Monterreal, P. M. Ortigosa, On building-up a yearly characterization of a heliostat field: A new methodology and an application example, Sol. Energy 173 (2018) 578–589.

R. Monterreal, A. Neumann, Advanced flux measurement system for solar tower plants, in: Proceedings of Seventh Intern. Symposium on Solar Thermal Concentrating Technologies, 1994, pp. 1–10.

D. J. Higham, N. J. Higham, MATLAB guide, SIAM, 2016.

I. Boussaïd, J. Lepagnot, P. Siarry, A survey on optimization metaheuristics, Inf. Sci. 237 (2013) 82–117.

G. Lindfield, J. Penny, Introduction to nature-inspired optimization, Academic Press, 2017.

R. V. Rao, Teaching–learning-based optimization algorithm, in: Teaching learning based optimization algorithm, Springer, 2016, pp. 9–39.

P. M. Ortigosa, I. García, M. Jelasity, Reliability and performance of UEGO, a clustering-based global optimizer, J. Glob. Optim. 19 (2001) 265–289.

D. Molina, M. Lozano, A. M. Sánchez, F. Herrera, Memetic algorithms based on local search chains for large scale continuous optimisation problems: MA-SSW-Chains, Soft Comput. 15 (2011) 2201–2220.

J. M. García-Martínez, E. M. Garzón, J. M. Cecilia, H. Pérez-Sánchez, P. M. Ortigosa, An efficient approach for solving the hp protein folding problem based on uego, J. Math. Chem. 53 (2015) 794–806.

N. C. Cruz, J. L. Redondo, J. D. Álvarez, M. Berenguel, P. M. Ortigosa, Optimizing the heliostat field layout by applying stochastic population-based algorithms, Informatica 29 (2018) 21–39.

F. J. Solis, R. J. B. Wets, Minimization by random search techniques, Math. Oper. Res. 6 (1981) 19–30.

J. L. Redondo, A. G. Arrondo, J. Fernández, I. García, P. M. Ortigosa, A two-level evolutionary algorithm for solving the facility location and design (1—1)-centroid problem on the plane with variable demand, J. Glob. Optim. 56 (2013) 983–1005.

M. A. Branch, A. Grace, MATLAB: Optimization Toolbox: User's Guide, Math Works, 2020.

C. P. López, Optimization techniques via the optimization toolbox, in: MATLAB Optimization Techniques, Springer, 2014, pp. 85–108.

R. H. Byrd, J. C. Gilbert, J. Nocedal, A trust region method based on interior point techniques for nonlinear programming, Math. Program. 89 (2000) 149–185.

N. C. Cruz, A compact MATLAB implementation of the TLBO optimizer, `https://github.com/cnelmortimer/TLBOSearch`, 2021a. Accessed: 2022-08-17.

N. C. Cruz, A compact MATLAB implementation of the UEGO optimizer, `https://github.com/cnelmortimer/UEGO`, 2021b. Accessed: 2022-08-17.

M. Matsumoto, T. Nishimura, Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator, ACM Trans. Model. Comput. Simul. 8 (1998) 3–30.