.

Ray-tracing software comparison for linear focusing solar collectors

Tiago Osório' , Pedro Horta, Marco Larcher, Ramón Pujol-Nadal, Julian Hertel, De Wet van Rooyen, Anna Heimsath, Simon Schneider, Daniel Benitez, Antoine Frein, and Alice Denarie

Citation: AIP Conference Proceedings **1734**, 020017 (2016); doi: 10.1063/1.4949041

View online: http://dx.doi.org/10.1063/1.4949041

View Table of Contents: http://aip.scitation.org/toc/apc/1734/1

Published by the American Institute of Physics

Ray-Tracing Software Comparison for Linear Focusing Solar Collectors

Tiago Osório[1, a)], Pedro Horta[1], Marco Larcher[2], Ramón Pujol-Nadal[3], Julian Hertel[3], De Wet van Rooyen[4], Anna Heimsath[4], Simon Schneider[5], Daniel Benitez[6], Antoine Frein[7] and Alice Denarie[7]

[1]*University of Évora – Renewable Energies Chair, Palácio Vimioso, Largo Marquês de Marialva, 7002-554 Évora, Portugal.*
[2]*Institut für Solartechnik SPF, Oberseestrasse 10, 8640 Rapperswil, Switzerland.*
[3]*University of the Balearic Islands (UIB), Physics Department, Ctra. Valldemossa km 7.52, 07122 Palma de Mallorca, Spain.*
[4]*Fraunhofer Institute for Solar Energy Systems, Division Solar Thermal and Optics, Heidenhofstraße 2, 79110 Freiburg, Germany.*
[5]*German Aerospace Center (DLR), Institute of Solar Research, Linder Höhe, 51147 Cologne, Germany.*
[6]*German Aerospace Center (DLR), Institute of Solar Research, Plataforma Solar de Almería, Spain.*
[7]*Politecnico di Milano (POLIMI), Energy Department, Lambruschini 4, 20156 Milano, Italy.*

[a)]Corresponding author: tiagoosorio@uevora.pt

Abstract. Ray-Tracing software tools have been widely used in the optical design of solar concentrating collectors. In spite of the ability of these tools to assess the geometrical and material aspects impacting the optical performance of concentrators, their use in combination with experimental measurements in the framework of collector testing procedures as not been implemented, to the date, in none of the current solar collector testing standards. In the latest revision of ISO9806 an effort was made to include linear focusing concentrating collectors but some practical and theoretical difficulties emerged. A Ray-Tracing analysis could provide important contributions to overcome these issues, complementing the experimental results obtained through thermal testing and allowing the achievement of more thorough testing outputs with lower experimental requirements. In order to evaluate different available software tools a comparison study was conducted. Taking as representative technologies for line-focus concentrators the Parabolic Trough Collector and the Linear Fresnel Reflector Collector, two exemplary cases with predefined conditions – geometry, sun model and material properties – were simulated with different software tools. This work was carried out within IEA/SHC Task 49 "Solar Heat Integration in Industrial Processes".

INTRODUCTION

Ray-Tracing (RT) software tools enable a straightforward assessment of optical performance aspects at collector or solar field levels and have been widely used in the framework of solar concentrator optical design: e.g., in the assessment of materials, in the evaluation of shading and blocking effects in solar fields or in the determination of energy flux distributions in central receiver systems.

In spite of such capabilities, the use of RT software tools within the solar collector certification process is not foreseen in any of the existing standards. Considering its general application to both stationary and tracking collectors, the most complete standard for solar collector testing is the latest revision of ISO9806:2013 [1], including already specific recommendations for the experimental procedures to be followed in the testing of tracking collectors. However in the effort of extending the previous procedures to this type of collectors some practical and theoretical difficulties emerge, notably:

SolarPACES 2015
AIP Conf. Proc. 1734, 020017-1–020017-8; doi: 10.1063/1.4949041
Published by AIP Publishing. 978-0-7354-1386-3/$30.00

- the need to tilt large collectors such as Linear Fresnel Collectors (LFC) to test under normal incidence conditions (whenever such tests do not occur at latitudes near 0°);
- the validity of the bi-axial (longitudinal and transversal) approximation for the Incidence Angle Modifier (IAM);
- the need to obtain the Optical End Loss factor separately (not included in the longitudinal IAM curve) for a collector that will be installed in rows of various lengths [2].

Whereas such aspects are not covered in the present version of the standard, it seems logical that the use of RT analysis could provide important contributions to overcome these issues, complementing the experimental results obtained through thermal testing and allowing the achievement of more thorough testing outputs with lower experimental requirements [3]. To do so, the first step is to understand which optical effects must be considered and included in the physical RT model [4] and then to distinguish RT tools according to the accuracy they provide in the simulation of such effects.

Within IEA/SHC Task 49 [5] a comparison between different RT software tools was conducted. Taking as representative technologies for line-focus concentrators the Parabolic Trough Collector (PTC) and the Linear Fresnel Reflector Collector, each participant was asked to describe their RT software regarding its features and capacities and then to run simulations of two exemplary cases, a PTC and a LFC with predefined conditions: geometry, sun model and material properties.

This article presents the results of this study and analyzes the differences between the RT tools including the different approaches to the simulation of some of the physical phenomenon with particular emphasis placed on refraction.

PARTICIPANTS AND RT SOFTWARE TOOLS IN THE STUDY

Six Task49 participants agreed to take part in the study: University of Évora (UEvora), Institut für Solartechnik (SPF), Universitat de les Illes Balears (UIB), Fraunhofer-Institut für Solare Energiesysteme (ISE), Deutsches Zentrum für Luft- und Raumfahrt (DLR) and Politecnico di Milano (POLIMI). Seven different RT tools were used in this comparison, including open source, commercial and in-house software. The PTC case was modeled by six of the RT tools and the LFC case by four, according to Table 1.

TABLE 1. Participants software tools and performed simulations

Participant	Software	Licence	Simulation
UEvora	Tonatiuh	Open-source	PTC, LFC
SPF	OptiCAD	Commercial	PTC
UIB	OTSun	In-house	PTC, LFC
ISE	Raytrace3D	In-house	PTC, LFC
DLR	STRAL**	In-house*	PTC
DLR	SPRAY	In-house*	PTC
POLIMI	SolTrace	Open-source	LFC

*copy available on license-fee
**experimental features for PTC RT

Software Presentation

Tonatiuh is a Monte Carlo ray tracing software implemented in C++ as a multiplatform open source project currently being developed by CENER. The simulation of one system is based in three models: the concentrating system model, an incoming solar radiation model and a model for interactions between the radiation and concentrating system elements. Tonatiuh allows the construction of a detailed optical system based on predefined geometric forms or by using surfaces generated by Bezier curves and the analysis of the optical behavior of each of the subsystems. It is optimized for parallel processing using all the available CPUs in order to reduce computation time. A user friendly graphical interface allows the visualization of the system and ray paths. At the moment, under the European Project STAGE-STE work package 12, new features are being developed including the data post processing, CAD import and angular dependence of material optical properties.

OptiCAD is a “100% non-sequential, illumination, optical analysis, and visualization program. OptiCAD can perform analysis on arbitrarily placed optical components, with the capability to do unconstrained ray tracing considering reflection, refraction, surface and bulk scattering surface and bulk luminance, and polarization. OptiCAD models many optical components, including lenses, mirrors, light pipes, prisms, imported CAD surfaces and solid objects, faceted surfaces and solid objects, and other components. Light sources may be modeled as points, lines, surfaces, volumes, bitmaps, equations, or data tables. Sources may be diverging converging, or collimated, and multiple sources may be placed at any location. Sources may also be user defined via table or external dlls.”

OTSun is a forward ray tracing tool based on the Monte Carlo method. The collector geometry is composed of triangles, which allows importing STL-files from common CAD programs. As typical for stochastical ray tracing programs, the trajectory of a ray starts from a randomly generated location within the sun window. In an iterative process intersection points with triangles of the object are determined and the trajectory is updated. In each intersected surface a new intensity value will be assigned to the beam depending on the material. Four different surface types can be defined, Opaque, Asorptive (with an incidence angle dependent probability), Specular and Refractive. Multiple reflections between two refractive layers is possible. To define the initial vector of the emitted beam, the sun model can be set to ideal (point source) or Buie model.

Raytrace3D is an in-house software suite that is used for the assessment and optimization of concentrating collectors. By continuous development it is adapted and constantly improved to fit the needs of project work. The simulation engine itself and the pre-/post-processing tools are implemented with a performant yet flexible combination of C/C++ and Bash/Python. The software suite has been successfully employed for various and very diverse applications. The core simulation engine is a quasi-Monte-Carlo forward ray tracer that considers an additional intensity value for every ray. Thus, material properties like reflectivity are either taken into account statistically or by adjusting the intensity of the ray. Performance is increased by bounding object algorithms that allow for a hierarchical clustering of the scene. The inherent autonomy of the rays allows for the parallel execution of the tracing process on multiple CPUs.

STRAL is a ray tracing software with graphical user interface written in C++ that has been developed at DLR (copy available on license fee). The main purpose of the software is the flux density simulation of heliostat fields with a very high accuracy in a small amount of computation time. The software is primarily designed to process real sun shape distributions and real highly resolved heliostat geometry data, which means a data set of normal vectors of the entire reflecting surface of each heliostat in the field. Specific receiver and secondary concentrator models as well as models of objects that are shadowing the heliostat field can be implemented by the user and be linked to the simulation software subsequently. The specific architecture of the software enables the provision of other powerful simulation environments (e.g. Matlab, Labview, Dymola, Excel) with precise flux density simulation data for the purpose of entire plant simulations. Rays are generated on the concentrator surface and SIMD inline assembly code is used to increase simulation speed. The capability of simulating parabolic trough concentrator technology has been implemented but is still in experimental state.

SPRAY is a command-line based ray tracing tool written in FORTRAN developed in-house at DLR (copy available on license fee). It is originally based on the code “MIRVAL” developed by Sandia National Laboratories for the evaluation of the optical performance of solar tower systems. The current version of SPRAY extends the original capabilities of MIRVAL extensively, like new concentrator geometries, evaluation methods and parameter, receiver models, interfaces to other software, etc. The basic principle of the ray tracing code is to generate a huge number of solar rays, and then trace them during their interaction with all components of the system under evaluation. Two basic methods for ray generation exist: ray generation on an ‘insolation box’ for forward ray-tracing as well as ray generation on the concentrator elements for backwards ray-tracing. A variety of material properties of the concentrator components (including shading and blocking) as well as ambient properties like sunshape, atmospheric attenuation etc. can be set within input files and evaluations can be done in single mode or batch mode or by calling SPRAY from within other software tools like Matlab.

SolTrace is a software tool developed at the National Renewable Energy Laboratory (NREL) to perform modelling and simulation of optical performances of concentrating solar collectors. The code (written in C++) is based on Monte Carlo ray-tracing methodology. In SolTrace, an optical system is organized into “stages” within a global coordinate system: it models optical geometries as a series of stages composed of optical elements with an extensive variety of available attributes including shape, contour, and optical quality. The software has a user interface to input and output data for analysis. After giving input data of geometry, optical material properties, optical errors and sun shape, the user selects a given number of rays to be traced. Each ray cross the system while encountering various optical interactions. Two kinds of interactions are evaluated: probabilistic ones for e.g. sun angle, optical errors and deterministic one for e.g. calculation of ray intersection with an analytically described surface and resultant redirection.

Software Capabilities

Information about the software capabilities was gathered comprising the sun model, real materials, surface errors, angular variation of optical properties and any other feature that the users think relevant. This information is compiled in Table 2 and Table 3.

TABLE 2. Software features – sun model, real material properties and surface errors

Software	Sun model	Real materials	Surface errors
Tonatiuh	Ideal; Pillbox; Buie	Yes	Specularity and slope with normal (univariate) or pillbox distributions
OptiCAD	Ideal; Lambertian; User defined	Yes	User defined or specularity and slope with gaussian, lambertian, power-law, or 2-fold exponential function (bivariate) distributions
OTSun	Ideal; Buie; Diffuse (isotropic sky)	Yes	Specularity and slope with normal (univariate) distribution
Raytrace3D	Ideal; Pillbox; Buie	Yes	Specularity and slope with normal (univariate/bivariate) or pillbox distributions
STRAL	Ideal; Pillbox; Normal Distribution; User defined	Only reflectors	Slope with normal (bivariate) distribution; User defined by an input external file
SPRAY	Ideal; Pillbox; Normal Distribution; User defined	Yes	Slope with normal (bivariate) distribution; User defined by an external input file
SolTrace	Pillbox; Gaussian; User defined	Yes	Specularity and slope with Gaussian (univariate) or pillbox distributions

All software tools can implement: Buie correlations for the sun model as an option in the program or as a user input; real material properties for reflective surfaces, refractive elements and absorber surfaces (except STRAL that only implements reflective surfaces); and both specularity errors and slope errors. Raytrace3D allows two different distributions for two different directions as well as OptiCAD, STRAL and SPRAY that can also use external files to define the errors.

TABLE 3. Software features – angular variation of optical properties and other capabilities

Software	Angular variation of optical properties	Other capabilities
Tonatiuh	No	Script files for designing and running simulations
OptiCAD	Yes	Optical properties can depend on the wavelength
OTSun	Yes	---
Raytrace3D	Yes	---
STRAL	No	Interface for tool coupling with Matlab, Labview, Dymola, Excel; Usage of heliostat surface data
SPRAY	Refraction: yes Absorption: no	Rays generated in "insolation box" or reflector surface; Operation by external software; Usage of heliostat surface data
SolTrace	Yes	Interface for tool coupling with google sketch up

OptiCAD, OTSun, Raytrace3D and SolTrace implement equations or user defined data so account for the angular variation of the optical properties while Tonatiuh and STRAL use the same values independent of the incidence angle. SPRAY deals with angular effects for refraction but not for absorption.

REFRACTION MODEL

To model refraction while some programs accept the transmission, absorption, reflection and refractive index values and apply Monte Carlo statistical approach over this values, other have the Fresnel equations implemented and don't accept all sets of values because transmissivity becomes a function of refractive index. To have the proper values for the simulation an anti-reflective (AR) coating must be modeled to achieve higher transmission. Also some software tools have an extinction coefficient which means that there is always absorption on the glass.

Tonatiuh: to model a refractive component on Tonatiuh one has to define two refractive surfaces with coefficients to the transmission, absorption, reflection and refractive index at each side. These values are constant no matter the incidence angle. Snell's Law is used to calculate the direction of the transmitted ray. As a consequence there is no decrease in transmission for incidence angles higher than zero.

OptiCAD: To model refraction and account for the angular dependence effects, an anti-reflective coating can be defined and the Fresnel equations applied.

Raytrace3D: In order to consider the effect of anti-reflection coatings, a set of two incident angle dependent reflectance curves are generated for the two sides of a glass to air/vacuum boundary. Measurements of incident angle dependent transmittance for an AR-coated glass plane serve as input. Furthermore the rays are refracted considering the refractive indices and Snell's law.

OTSun: To model refractive material such as glass, the ray tracing software considers reflection and transmission probabilities according to the electromagnetic Fresnel equations as well as extinction according to Beer-Lambert. The glass is completely described when specifying the following parameters at the respective interface: refractive index of air, refractive index of glass, extinction coefficient of glass. To model an AR coating an additional layer was added with a refractive index of $\mathrm{n_{coating}} = \sqrt{\mathrm{n_{glass}}}$ [6].

SPRAY: A glass tube is implemented as a "blocking element". Optical characteristics of the glass envelope are defined by: index of refraction of glass, absorption coefficient, cover thickness and relative loss factor from dirt. Reflectivity at the cover glass is calculated with Fresnel equations. Propagation into glass is calculated with Snell's law (refraction). Absorption in glass is based on path length in glass.

SolTrace: SolTrace deals with refractive or reflective components in two different ways. For refractive components just transmission and reflection are considered while for the reflective one only the reflection is considered. For transmission there is the possibility to define transmission coefficients (two values for two interfaces of the material) and the index of refraction. In this case the Fresnel equation is handling the reflection and refraction at surface interfaces (angular variation). For reflection the angular variation is handled by an external table that should be defined which allows the user to enter values for reflectivity as a function of the incidence angle.

OPTICAL SURFACE ERRORS

It is generally assumed in ray tracing simulation models that surface errors follow a Gaussian [7] distribution characterized by the standard deviation σ. The total error accounting for all surface irregularities consists of the angle between the ideal specular and deviated reflection direction, expressed in terms of the reflected vector as:

$$\theta_{TOT,surf} = 2\theta_{slope} + \theta_{spec} + \theta_{track} + \theta_{align} \tag{1}$$

$\theta_{TOT,surf}$ – Surface error: the global error of the surface
θ_{slope} – Slope (contour) error: irregularities in the shape of the reflector
θ_{spec} – Specularity (dispersion, spread beam) error: microscopic irregularities of the surface
θ_{track} – Tracking error: irregularities in reflector focusing on the receiver
θ_{align} – Alignment error: displacement from correct position due to imperfect installation, wind or soiling [8]

The multiplication by 2 in the first term is caused by Snell law, as θ_{spec}, θ_{track} and θ_{align} are referred to the reflected ray on the receiver while θ_{slope} is referred to the surface normal (deviation of θ on the surface normal causes a deviation of 2θ on the direction of the reflected ray on the receivers). The standard deviation $\sigma_{tot,surf}$ is:

$$\sigma_{TOT,surf} = \sqrt{\left(2\sigma_{slope}\right)^2 + \sigma_{spec}^2 + \sigma_{track}^2 + \sigma_{align}^2} \tag{2}$$

Differently from PTC, reflectors and receiver do not move together in the case of LFC, but separately and the error is applied to the normal direction of the mirror, focusing on the receiver. In LFC systems the total error is:

$$\sigma_{TOT,surf} = \sqrt{\left(2\sigma_{slope}\right)^2 + \sigma_{spec}^2 + \left(2\sigma_{track}\right)^2 + \sigma_{align}^2} \tag{3}$$

Most of the RT tools only allow for slope and specularity errors to be defined or even only slope error. If just the slope error has to account for all effects, in order to have the $\sigma_{TOT,surf}$ desired it can be considered a virtual $\sigma_{slope} = \sigma_{TOT,surf}/2$ after equations 2 and 3.

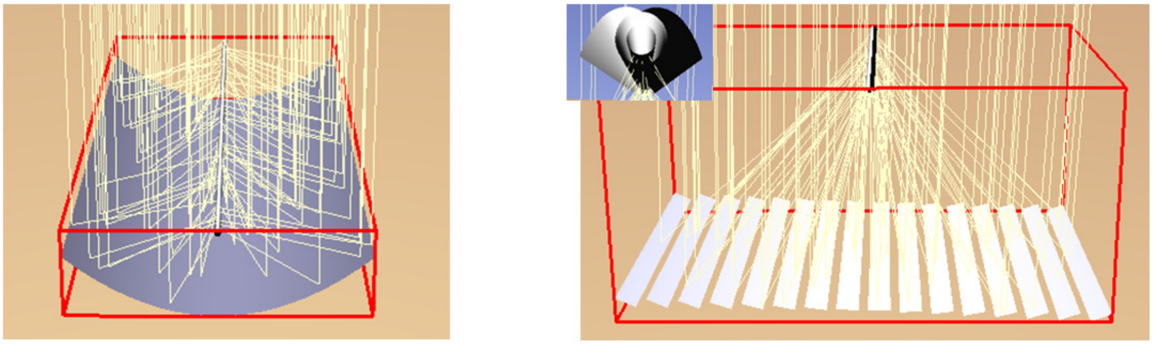
DEFINITION OF THE SIMULATION CASES

The reference area was defined as the horizontal projection, at normal incidence, of the directly irradiated primary reflector and receiver areas. The two simulation cases in the comparison were defined according to the conditions presented in table 4.

TABLE 4. Definitions of the simulated cases

	PTC	**LFC**
Geometry	5.8 m width parabola 1.71 m of focal length	16 parabolic heliostats (0.75 m wide) 7.4 m height
Secondary	---	CPC: $\theta_a = 48.39$ º; $h_t = 41$ mm
Receiver evacuated tube	35 mm absorber radius 62.5 mm outer radius and 5 mm thickness glass tube	
Collector length	12 m	
Materials (normal incidence)	reflector: $\rho = 0.935$; absorber: $\alpha = 0.955$; glass: $\rho = 0.035$; $\tau = 0.965$; $n = 1.52$	
Slope deviation (for reflectors)	$\sigma_s = 2.5$ mrad	
Reference clear day sky	Buie 5%	

The outcome of the simulations to be compared was the efficiency at normal incidence η_0, the longitudinal (L) IAM curve for the PTC and the bi-axial (longitudinal and transversal (T)) IAM curves for the LFC.



(a) (b)

FIGURE 1. PTC (a) and LFC with a detail view of the secondary (b) simulation cases – images from Tonatiuh.

RESULTS

Table 5 summarizes the options taken in each simulation limited by the software option to follow the proposed simulation conditions.

TABLE 5. Simulation conditions for each software

Software	**Sun model**	**Materials**	**Reflector surface error**	**Angular variation of optical properties**
Tonatiuh	Buie 5%	Real	Univariate normal distribution	No
OptiCAD	3 mrad Gauss	Real	Univariate normal distribution	Yes
OTSun	Buie 5%	Real	Univariate normal distribution	Yes
Raytrace3D	Buie 5%	Real	Univariate normal distribution	Yes

STRAL	Buie 5%	Absorber: α = 1 No glass tube	Bivariate normal distribution	No
SPRAY_1	Buie 5%	Absorber: α = 1 No glass tube	Bivariate normal distribution	No
SPRAY_2	Buie 5%	Real (with n = 1)	Bivariate normal distribution	No
SPRAY_3	Buie 5%	Real (no AR)	Bivariate normal distribution	Yes
SolTrace	3 mrad Gauss	Real	Univariate normal distribution	Yes

Optical Efficiency Results

TABLE 6. Optical efficiency for the PTC case

Tonatiuh	OptiCAD	Raytrace3D	OTSun	SPRAY_2	SPRAY_3	Mean	STRAL	SPRAY_1
0.843	0.858	0.828	0.830	0.846	0.805	0.835	0.918	0.918

The PTC optical efficiency for the proposed materials should have a maximum value of (ρτα) = 0.862 which corresponds to the situation of an ideal point source and no surface errors. The effect of the real sun model and surface errors decreases slightly this value. In the case of SPRAY_1 and STRAL the values are higher since there is no glass and the absorber is perfect. In SPRAY_2 it is shown the effect of introducing the glass but only modelled with a transmission coefficient without considering the refractive index. This value is very similar to the one obtained in Tonatiuh which only adds the glass refractive index. OptiCAD (with a 3 mrad Gaussian as sun model), Raytrace3D and OTSun implement an AR coating to achieve the desired transmissivity with the given refractive index. SPRAY_3 includes an absorption coefficient in the glass but no AR coating is used.

TABLE 7. Optical efficiency for the LFC case

Software	Tonatiuh	Raytrace3D	OTSun	SolTrace	Mean
η_0	0.734	0.721	0.756	0.738	0.737

For the LFC case Tonatiuh and SolTrace (with a 3 mrad Gaussian as sun model) results are similar since the glass transmissivity is imposed as an input. Raytrace3D and OTSun have an AR coating model. The differences in the results should be explained by the differences in modeling the glass tube or the secondary shape.

Incidence Angle Modifier Results

To compare the simulation results for the IAM, the absolute differences from the mean value were calculated for each incidence angle value. In fig. 2 (PTC) and fig. 3 (LFC) the values are presented for 10º, 30º, 50º and 70º incidence angles.

While Tonatiuh, SPRAY_1, SPRAY_2 and STRAL are unable to account for angular variation of optical properties, OptiCAD, Raytrace3D, OTSun and SPRAY_3 have diverse ways of dealing with this effect both in the absorber as in the glass. The resulting differences (fig. 2) increase with the value for the incidence angle.

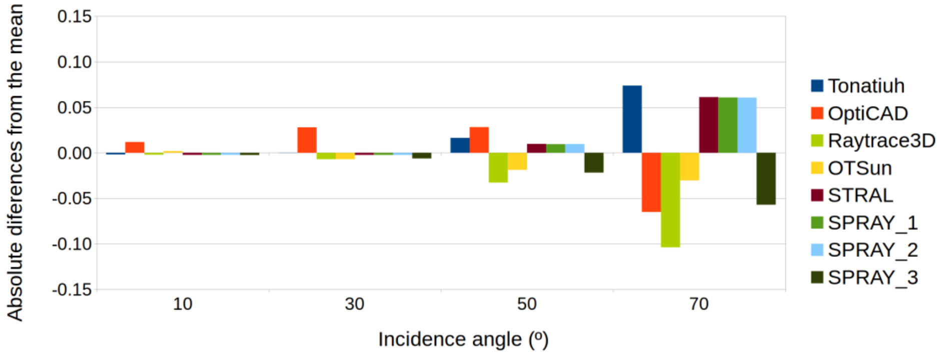


FIGURE 2. IAM absolute differences from the mean for the PTC case.

For the LFC case, the differences are smaller for the analyzed software tools (fig. 3). Here the dominant effects on the efficiency are geometrical: the use of a secondary reflector minimizes the impact of differences in the reflected rays and the large end losses on the longitudinal direction makes high incidence angles on the glass tube impossible. The deviations observed in the PTC case are then hidden.

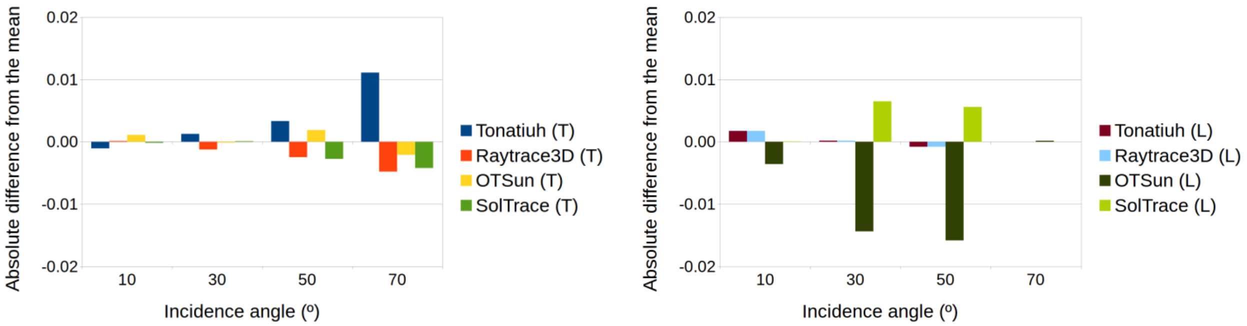


FIGURE 3. Transversal (T) and Longitudinal (L) IAM absolute differences from the mean for the LFC case.

CONCLUSIONS

When comparing the software capabilities the main limitations of each tool were noticed. Besides one of the tools that cannot model a glass element, the main differences were in the degree with which each software could model the angular dependency of the material optical properties. In particular refraction on the glass tube is not modeled in the same way by the different tools. Although good agreement was obtained, it is important to notice that the different modeling options by different software tools produce different optical efficiency values and IAM curves.

A deeper look into the reasons of some of the observed differences and a sensitivity analysis on the impact of the simulation parameters in the final results are the next step. From this study a common guideline can be proposed to classify a RT software tool as adequate, and to which level of detail, to model a linear concentrating solar collector and to which extent RT simulation results can be used as an auxiliary instrument to the experimental standardized thermal tests of linear concentrating collectors.

ACKNOWLEDGMENTS

T. Osório gratefully acknowledges the Portuguese funding institution FCT – Fundação para a Ciência e a Tecnologia – for supporting his research.

REFERENCES

1. ISO 9806:2013, Solar energy – Solar thermal collectors – Test methods (2013).
2. A. Heimsath, G. Bern, D. van Rooyen and P. Nitz, “Quantifying optical loss factors of small linear concentrating collectors for process heat application”, Energy Procedia **48**, 77-86 (2014) .
3. R. Pujol, “Optical and thermal characterization of a variable geometry concentrator using ray-tracing tools and experimental data”, Applied Energy **155**, 110–119 (2015).
4. P. Horta, T. Osório, “Optical characterization parameters for line-focusing solar concentrators: Measurement procedures and extended simulation results”, Energy Procedia **49**, 98-108 (2014).
5. International Energy Agency Solar Heating and Cooling Programme, Task 49 “Solar Heat Integration in Industrial Processes”.
6. A. Gombert, W. Glaubitt, K. Rose, J. Dreibholz, B. Bläsi, A. Heinzel, D. Sporn, W. Döll and V. Wittwer, “Antirefractive transparent covers for solar devices,” Sol. Energy **68**, 357–360 (2000).
7. R. B. Pettit, “Characterization of the Reflected Beam Profile of Solar Mirror Materials”, Sol. Energy **19(6)**, 733–741 (1977).
8. G. Zhu, “Development of an analytical optical method for linear Fresnel collectors”, Sol. Energy **94**, 240-252 (2013).