

Manuel Blanco¹

Energy, Environment and
Water Research Center (EEWRC),
The Cyprus Institute,
Athassa Campus,
20 Konstantinou Kavafi Street,
Aglantzia,
Nicosia 2121, Cyprus
e-mail: m.blanco@cyi.ac.cy

Marios Constantinou

Energy, Environment and
Water Research Center (EEWRC),
The Cyprus Institute,
Athassa Campus,
20 Konstantinou Kavafi Street,
Aglantzia,
Nicosia 2121, Cyprus
e-mail: m.constantinou@cyi.ac.cy

Clotilde Corsi

Research School of Engineering,
Australian National University,
Canberra ACT 0200, Australia
e-mail: clotilde.corsi@csiro.au

Victor Grigoriev

Energy, Environment and
Water Research Center (EEWRC),
The Cyprus Institute,
Athassa Campus,
20 Konstantinou Kavafi Street,
Aglantzia,
Nicosia 2121, Cyprus
e-mail: v.grigoriev@cyi.ac.cy

Kypros Milidonis

Energy, Environment and
Water Research Center (EEWRC),
The Cyprus Institute,
Athassa Campus,
20 Konstantinou Kavafi Street,
Aglantzia,
Nicosia 2121, Cyprus
e-mail: k.milidonis@cyi.ac.cy

Constantinos F. Panagiotou

Energy, Environment and
Water Research Center (EEWRC),
The Cyprus Institute,
Athassa Campus,
20 Konstantinou Kavafi Street,
Aglantzia,
Nicosia 2121, Cyprus
e-mail: c.panagiotou@cyi.ac.cy

Costas N. Papanicolas

Energy, Environment and
Water Research Center (EEWRC),
The Cyprus Institute,
Athassa Campus,
20 Konstantinou Kavafi Street,
Aglantzia,
Nicosia 2121, Cyprus
e-mail: cnp.energy@cyi.ac.cy

John Pye

Research School of Engineering,
Australian National University,
Canberra ACT 0200, Australia
e-mail: john.pye@anu.edu.au

FluxTracer: A Ray Tracer Postprocessor to Assist in the Design and Optimization of Solar Concentrators and Receivers

This paper presents FluxTracer, an advanced open source computer tool to assist in the analysis, design, and optimization of solar concentrators and receivers. FluxTracer is a postprocessor for Monte Carlo ray tracers used to simulate the optical behavior of solar concentrating systems. By postprocessing the rays generated by the ray tracer, FluxTracer can partition into volumetric pixels (voxels) a region of interest in three-dimensional (3D) space defined by the user and compute for each voxel the radiant power density of the concentrated solar radiation. Depending upon the set of rays provided by the ray tracer, it may be able to integrate the radiant power density in every voxel over time. The radiant energy density analysis described is just one of the analyses that FluxTracer can carry out on the set of rays generated by the ray tracer. This paper presents the main analyses that FluxTracer can provide. It also presents examples of how the information provided by FluxTracer can be used to assist in the analysis, design, and optimization of solar concentrators and receivers. FluxTracer is the first of a series of components of an open-source computational framework for the analysis, design, and optimization of solar concentrators and receiver, being developed by The Cyprus Institute (CyI) and the Australian National University (ANU). [DOI: 10.1115/1.4042127]

¹Corresponding author.

Contributed by the Solar Energy Division of ASME for publication in the JOURNAL OF SOLAR ENERGY ENGINEERING: INCLUDING WIND ENERGY AND BUILDING ENERGY CONSERVATION. Manuscript received September 27, 2018; final manuscript received November 20, 2018; published online January 8, 2019. Guest Editors: Tatsuya Kodama, Christian Sattler, Nathan Siegel, Ellen Stechel.

1 Introduction

The current world energy system is not sustainable. It is mainly based on technologies that use finite resources, have a large carbon footprint, and are environmentally unfriendly. The transformation of this energy system into one that only uses renewable energy sources and which, therefore, has a negligible carbon footprint and is environmentally friendly and fully sustainable, is one of the main challenges of our times. This is the case, not just because of the sheer magnitude of the endeavor, but also because of the urgency with which this transformation must be undertaken if we are to avoid a major environmental disaster.

In the new world energy system, toward which we must evolve rapidly, concentrating solar thermal (CST) technologies are anticipated to play an important role, because of their capacity to incorporate thermal energy storage, to be hybridized with conventional and renewable energy sources, to provide ancillary services to energy grid, and many other advantages. This is increasingly being recognized in many analyses [1] and in the energy policies of countries, which are actively trying to decarbonize rapidly their energy systems.

To ensure that CST technologies fulfill their promise to play a relevant role in the new world energy system, however, it is essential to increase substantially their cost-competitiveness. This performance to cost ratio can be improved by improving performance or decreasing cost, or any combination of these including, in some cases, small increases in cost if performance gains are enough.

For all commercial CST technologies, the light collection and concentration (LCC) subsystem is, together with the power block, the subsystem that has the greatest influence in cost-competitiveness: its optimization is critical. This subsystem is composed of the surfaces that collect and concentrate the sunlight and of the input surfaces of the receivers or of the receivers' envelopes, where the light is concentrated.

The optimization of the LCC subsystems requires, in many cases, the optimization of the position, geometry, and size of a very large number of solar collecting and concentrating surfaces (heliostats, typically) as well as the optimization of the shape and size of the absorbing surfaces of the receivers upon which the sunlight is concentrated. This is especially the case in solar tower technology. Because a full optimization requires the exploration of a parameter space with a very large number of dimensions, the traditional optimization approach consists of making numerous simplifying assumptions to reduce drastically the parameter space to a tractable number, so that optimization can be carried out using conventional high-end workstations in a matter of hours.

To achieve breakthroughs, which substantially increase the cost-competitiveness of CST systems, a bolder approach may be needed, such as sophisticated design and analysis tools, engineered from the start to take advantage of any available multiprocessing capabilities and to run in high-performance computing (HPC) systems. Tools that can be combined with sophisticated optimization strategies, to explore and find optimal solutions in very high dimensional configuration spaces.

This paper presents the first of a series of such design and analysis tools. The tool, called FluxTracer, is a Monte Carlo ray tracing (MCRT) postprocessor, which partitions the three-dimensional (3D) space occupied by the LCCS into volumetric pixels (voxels) and postprocesses the MCRT-generated rays to compute the radiant energy flux that traverses each voxel as a function

of time. It integrates the power density in every voxel overtime, providing detailed information regarding how the radiant energy flows in space in a given LCCS and in a given period. FluxTracer is just one of several components that will be integrated in an open source computational framework for optimization of LCCSs, under development by the Cyprus Institute (CyI) and the Australian National University (ANU), using HPC. Further components are planned, but initial work has been focused on this one due to its simplicity and potential interest from the research community.

In what follows, the use of HPC, including parallel computing, by the CST research community is reviewed to contextualize the research effort presented here. The design goals and development approach of FluxTracer are described. The status of FluxTracer and some preliminary results are presented. Finally, the lessons learned, the conclusions reached, and the future work are discussed.

2 Parallel Computing and High Performance Computing in the Context of Concentrating Solar Thermal Research

Many of the problems associated with the development of CST technologies and their commercial deployment could benefit from the use of parallel computing techniques and of HPC facilities.

In relation to the solar resource available to drive CST systems, HPC facilities are instrumental for detailed modeling of the Earth atmosphere using numerical weather prediction (NWP) models; these models serve as the basis for solar direct normal irradiance (DNI) assessment, forecasting, and now casting [2]. HPC facilities are also instrumental for the use of numerical techniques to enhance the spatial resolution of NWP models and to use them to generate historical series of DNI data of relatively high-temporal resolution for the generation of synthetic typical meteorological year data sets at any point on Earth. Computationally intensive NWP models can also be combined with aerosol measurements, satellite data, and an accurate local modeling of the atmosphere at the location of a CST system to obtain real-time estimates of DNI and atmospheric transmissivity.

In relation to the simulation, design, and optimization of the LCCS of CST systems, HPC is instrumental, for instance, to explore systematically the usually very large number of possible combinations of parameters that influence the design of such a system. They are also influential in enabling solar researchers to experiment with optimization algorithms that, although computationally intensive, have the potential to improve the state of the art of LCCS [3,4]. This is especially relevant, as mentioned earlier, for the LCCS of solar tower systems, given the large number of individual heliostats, whose location in the field and aiming strategy must be optimized.

In relation to the simulation, design, and optimization of the different thermal components of CST systems (receivers, thermal storage, heat exchangers, turbo-machinery, etc.), parallel computing and access to HPC facilities are instrumental, for instance, to explore systematically the configuration space to optimize the systems [5]. They are also instrumental to carry out detailed computational fluid dynamics analysis and integrate them within the optimization loop [6,7].

In relation to the simulation of the overall behavior of the CST system and the optimization of its operation, parallel computing

and access to HPC facilities are instrumental, for instance, to improve the level of detail and the time span at which the overall simulation of CST system can be carried out. They are also instrumental to approach the optimization of the CST system as an overall optimization problem that seeks to optimize the overall annual performance of the system with which a higher degree of accuracy is usually customary [8–10].

In recent years, ray-tracing applications and particularly MCRT applications have gained popularity to simulate the optical behavior of CST technologies. In addition, some work has been also performed on using discrete ordinates in computational fluid dynamics to do analysis similar to MCRT [11,12]. This has been facilitated by parallel computing, which has decreased substantially the time required to solve problems of interest using this computationally intensive approaches. For example, the flux density distribution generated on the surface of a central receiver by a heliostat field has been estimated with high accuracy in few seconds using highly parallelized MCRT algorithms specially designed to run on graphics processing units.

Within the Australian Solar Thermal Research Initiative (ASTRI), the Australian National University (ANU) is leading the development of a series of open-source tools for overall simulation and optimization of CST systems, specifically solar tower systems. This is done under the framework of the SolarTherm project [13]² with the use of the Modelica language. Some of these tools are useful for the design and optimization of the heliostat fields and can be easily parallelized.

An open-source MCRT tool for the simulation of the optical behavior of solar thermal power systems, *Tonatiuh*, implemented multiprocessing capabilities almost from its inception more than 10 years ago [14]. The system advisor model [15], implemented by the National Renewable Energy Laboratory (NREL) and which was recently released as open-source code, implements multiprocessing calculations for optimization and sensitivity analysis of CST power plants.

Moreover, Tracer, a ray-tracing engine and suite of tools focused on solar energy application, developed by ANU, has multiprocessing capability as of recently, and its use in optimization studies on the National Computational Infrastructure supercomputer is increasing. For example, ANU recently conducted a comprehensive parameter search, using a parallel-processing task queue, to find configurations of the CSIRO solar field that would best mimic the flux conditions of the heliostat field of the PS10 solar tower power plant [16].

3 FluxTracer Design Goals and Computational Paradigm

The main purpose of FluxTracer is to provide the user with the possibility of carrying out diverse types of analysis of the solar radiation on a region of interest within a solar concentrator, such as a parabolic trough, a parabolic dish, or a heliostat field. These analyses are carried out by postprocessing the large set of rays generated by modeling and simulating the optical behavior of the solar concentrator of interest with a MCRT. Currently, FluxTracer is able only to read rays in the format provided by the MCRT Tonatiuh [17]. However, the rays FluxTracer processes may not be generated by this specific MCRT, other MCRT can be used, if the rays they generate can be input into FluxTracer in Tonatiuh's format. In the future, it is expected that FluxTracer will be able to handle other MCRT ray formats. Furthermore, we intend to integrate within Tonatiuh the current functionalities provided by FluxTracer, so that when using Tonatiuh we will be able to perform the FluxTracer type of calculations “on-the-fly,” without the need to save the rays to disk and read them from the disk.

Currently, the main functionalities of FluxTracer are the following:

- *Radiant density analysis*: In this analysis, FluxTracer computes the density distribution of radiant energy within the region of interest during a specified time interval.
- *Point analysis*: In this analysis, given a user-defined point within the region of interest and a set of rays provided by the MCRT, FluxTracer computes, for each one of the rays in the set, the minimum distance from the ray to the user-defined point and the location in space where this minimum distance takes place. These results can be used in sizing receivers and assessing the overall performance of LCC subsystem.
- *Line analysis*: Similar to point analysis, but with the user-defined line playing the role of the user-defined point. Thus, in this analysis given a user-defined line within the region of interest and a set of rays provided by the MCRT, FluxTracer computes, for each one of the rays in the set, the minimum distance from the ray to the user-defined line and the location in space where this minimum distance takes place. As in the previous analysis, these results can be used in sizing receivers and assessing the overall performance of LCC subsystem.

As already stated, FluxTracer is the first of a series of software tools, which are part of an open source software framework that CyI and ANU are developing to assist in the analysis, design, and optimization of CST systems. This framework is engineered from the start to run efficiently on HPC clusters.

As part of this CST analysis, design, and optimization framework, the design goals of FluxTracer are for the program to be

- applicable to all solar concentrating geometries of interest,
- able to harness, in a seamless way, all computational power available to it, including access to HPC,
- easy to use and maintain,
- able to accept input data from a variety of optical analysis Monte Carlo ray tracer programs, and
- able to provide results in a variety of formats, which can be readily processed by other programs to assist in the analysis and presentation of results.

Since FluxTracer is a MCRT postprocessor, its computational paradigm is that of classical MCRT program, such as Tonatiuh. In this paradigm, the flow of radiant energy within the optical system is represented by the concept of a ray. Rays obey the Eikonal equation [18]. They are perpendicular to the wave-front and transport the radiant energy. They bounce within the optical system following the laws of geometrical optics. Under this paradigm, the amount of radiant energy that traverses a given region of space in a given amount of time is proportional to the fraction of rays entering the optical system that traverses the given region of space.

How the mentioned analyses are carried out, and especially how they are streamlined, so that the number of rays that must be generated by the MCRT program and postprocessed by FluxTracer to provide good estimates is minimized, is not a trivial task. The modeling of atmospheric transmissivity also requires careful considerations.

4 Development Approach

Researchers at both the Energy Division of CyI and the Solar Thermal Group of ANU are fully convinced that research groups should openly share the software they develop and allow others full access to the source code so that they can further enhance it, identify and correct errors in the code, and contribute to its further development. Both organizations are committed to contributing to unleash the creativity of the CST research community and the public at large by

- developing advanced computer tools for the analysis and design of LCC subsystems of CST systems in general and particularly of solar tower systems, and

²<https://github.com/solartherm>

- making them fully open source and available to anyone interested in using and/or improving them.

Because of their above shared belief, CyI and ANU have agreed to make and treat FluxTracer from the start as a fully fledged C++ open source program. Thus, as soon as the first fully functional version of the program will be released, it will be hosted in a public Git repository and made available to anyone interested in using or modifying the code. Furthermore, a conscious effort is being made to minimize the program's dependencies on external libraries. This is expected to facilitate the setup of the programming environment needed to contribute to the development of the program and, therefore, to facilitate contributions to the program from the CST research community and the public at large. The goal is that FluxTracer as well as the rest of advanced computer tools that CyI and ANU is developing to assist in the analysis, design, and optimization of CST systems will not be the property of any individual or institution but the common property of the CST research community as a whole.

Currently, FluxTracer is being developed in two "versions":

- FluxTracerCPP17, based solely on the C++ 2017 standard with no dependencies on external third-party libraries,
- FluxTracerQt5, based on the C++ 2011 standard and heavily dependent on the Qt-5.11 library.

Both FluxTracer versions are console applications and should provide the same functionality. However, FluxTracerQt5 has currently implemented more functionality than FluxTracerCPP17. Because of this, the former version has been run to produce all the results presented in this paper.

The reason to develop the program in two different versions is that each approach has advantages and disadvantages. It is not clear, a priori, which version will deliver better performance when used in HPC environments and which will be easier to maintain and to modify. The advantage of FluxTracerCPP17 is that it does not depend on third party libraries, only on the pure C++ computer language. To achieve this elegantly, the latest version of the C++ language, C++ 2017, was chosen, but this was found to be rather problematic, due to still rather limited support for the C++ 2017 standard in available compilers, particularly the MinGW compiler.

In both versions, FluxTracer is targeted to run on the CyI and ANU HPC facilities, or any other similar facility. The initial test of FluxTracer is being performed at the CyI HPC facility, "CyTera," which is Linux based. This facility consists of 116 nodes, each with 12 cores and 48 GB of memory. Each node has two hexa-core sockets with Intel Westmere X5650 processors, each having 12 MB Cache, a speed of 2.66 GHz, and an Intel QPI bus with a speed of 6.40 GT/s.

Currently, FluxTracer runs on the mentioned HPC cluster using a bash script. The script creates and runs instances of the program in as many nodes as requested by the user or allowed by the system. For instance, if 52 points have to be processed for an annual simulation, and four nodes are allocated to perform the calculations, then each node is assigned to process 13 points. Of course, the number of points that each node is able to process simultaneously is dependent on the computational performance of the node. It is envisioned, however, that future versions of the program will run in a less embarrassingly parallel mode [19] and will make use of open message passage interface to implement more advanced and efficient parallelization.

The development of FluxTracer is being carried out using a relatively eclectic SCRUM/agile project management approach based on weekly "sprints." For each of the two versions of the program, the sprints alternatively focus on enhancing the program functionality and improving the program architecture or the quality of the code. Every two sprints a different program version is the focus of the development, and at the end of the second sprint, typically, an updated version is released. For the time being, all these program releases are private early stage preliminary releases. The public open source release is planned for the first quarter of 2019.

5 Status

A proof-of-concept of FluxTracer has been developed in the two versions, FluxTracerCPP17 and FluxTracerQt5, as noted earlier. The main input data to FluxTracer in both cases are the set or sets of rays generated by the MCRT program Tonatiuh in which the CST system of interest should be modeled, and its optical behavior simulated for the specified times and solar conditions to be analyzed. As stated earlier, currently, FluxTracer requires rays in the format exported by Tonatiuh. However, it is envisioned that future releases of the program will be able to import rays from other MCRT tools in other formats. Since the program postprocesses the rays generated from the MCRT, to ensure the quality of FluxTracer estimates, it is critical that the set or sets of rays provided be representative of the optical behavior of the solar concentrator under analysis.

The two FluxTracer versions have the same functionality. They can carry out radiant density analysis, point analysis, and line analysis. Sections 5.1–5.3 describe the current implementation of these functionalities.

5.1 Radiant Density Analysis. In this analysis, FluxTracer computes the density distribution of radiant energy within the region of interest during a specified time interval. The starting point is the sets of rays generated by the MCRT, which are representative of the optical behavior of the solar concentrator during the interval of time considered.

For the radiant density analysis, the user should specify the following information:

- (1) The size and location of the three-dimensional region of interest. In the current implementation, this is a cuboid with the sides aligned with the axes of the global coordinate system used in the MCRT to model and simulate the solar concentrator being analyzed. The position and dimensions of the cuboid are defined by the minimum (x_{\min} , y_{\min} , z_{\min}) and maximum (x_{\max} , y_{\max} , z_{\max}) coordinates of its corners.
- (2) The number of subdivisions (n_x , n_y , n_z) for the region of interest in each of the three directions parallel to the axes of coordinates.
- (3) The set of MCRT rays to be processed by FluxTracer.

The first two items determine how the region of interest is defined and voxelized. Obviously, there must be a proportionality between the number of voxels defined within the region of interest (i.e., the voxel resolution) and the number of rays being processed. There are also obvious tradeoffs between the size of the region of interest, the voxel resolution, and the number of rays.

Based on the input data, FluxTracer defines a three-dimensional array of dimension (n_x , n_y , n_z) in which each element contains the sum of the radiant energy contributed by all the rays that pierce the corresponding voxel when they are traversing the region of interest. When all the rays have been processed, the information contained in the tensor is transformed to radiant energy density by dividing their radiant energy content by the voxel volume. Since a key element of this functionality of FluxTracer is the voxel traversal algorithm, a substantial effort has been devoted to identifying a fast and computationally efficient one. For the proof of concept, the classical voxel traversal algorithm of Amanatides and Woo [20] has been implemented in C++.

5.1.1 Solar Radiant Energy Density Distribution in the Focal Region of Solar Concentrators. A straightforward application of the radiant density analysis is to study the distribution of radiant energy near the focal point of a solar concentrator. To carry out this analysis, there should not be a receiver or any other object located in the region. Unfortunately, Tonatiuh is used to simulate a solar concentrator, the program, to save computational time, will not cast a reflected ray unless it intersects an object. Thus, to ensure that Tonatiuh still generates rays that pass through the focal region, after eliminating the objects, which could intersect

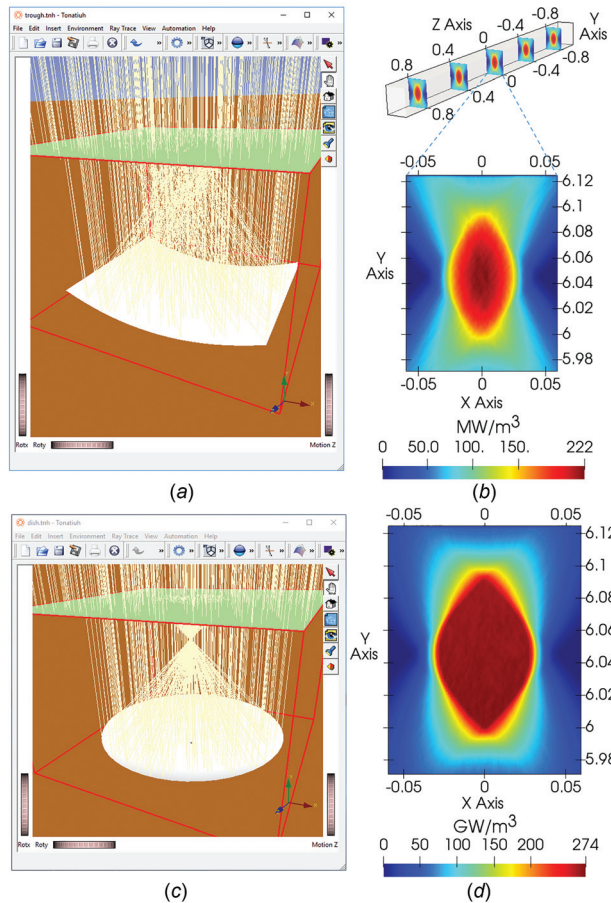


Fig. 1 Examples of how to simulate solar concentrators in Tonatiuh to generate rays crossing the focal region for (a) a parabolic trough and (c) a paraboloid dish. The key is the addition of a virtual rectangle above the focal region. Results of the analysis with FluxTracer are shown in (b) and (d), respectively.

rays within that region, one must add a virtual surface outside the region to intersect the rays crossing the region. The virtual surface could be just a flat rectangle appropriately sized and oriented as to capture, i.e., “intersect,” any ray passing through the focal region.

Figures 1(a) and 1(b) show the Tonatiuh models used to analyze the radiant energy distribution around the focal line of a parabolic trough and a parabolic dish concentrator, respectively. The exact geometry of these two concentrators and the main parameters used in Tonatiuh to generate the set of rays postprocessed by FluxTracer are presented in the section of this paper about the preliminary validation of FluxTracer. Figures 1(c) and 1(d) show the cross section in the x - y plane of the corresponding radiant energy densities around the focal line/point, estimated in MW/m^3 and GW/m^3 , respectively. Typically, the numbers of rays processed are in excess of 50×10^6 , and the number of voxels defined within the region of interest in excess of 729,000 (90, 90, 90).

5.1.2 Solar Radiant Energy Density Distribution Inside a Cavity Receiver. A more sophisticated application of the radiant density analysis functionality is to study the distribution of the radiant energy density inside a cavity receiver. As an example, we have carried out such an analysis for iStore receiver developed by CyI. This is a 150-kWth receiver designed by CyI as part of the proof of concept of a concentrating solar power and desalinated sea water (CSP-DSW) plant being developed at the CyI [21–26]. Figure 2 shows a view of the iStore receiver in operation on top of the tower of the CSP-DSW proof-of-concept at the CyI’s research and testing facilities at Pentakomo, Cyprus.



Fig. 2 The Cyprus Institute’s “iStore” receiver in operation at the research and testing facilities of the Institute at Pentakomo, Cyprus

To carry out the analysis, the geometry of the cavity of the iStore receiver, which is relatively complex, was modeled and imported as a 3D computer aided design object into Tonatiuh. The optical behavior of the surfaces of the cavity was modeled as “specular rough standard material” using a reflectivity of 30% and sigma slope of 45 deg, which is the standard deviation of the surface normal. Figure 3 shows the iStore cavity receiver and the heliostat field of the CSP-DSW facility being modeled and simulated in Tonatiuh. The instant of time simulated was the summer solstice at noon.

Over 6×10^9 rays were cast from the sun; of these, 314.6×10^6 reached the region of interest. This region was defined as a cube of $2 \text{ m} \times 2 \text{ m} \times 2 \text{ m}$ centered at the focal point of the heliostat field, placed 13.83 m above the ground. This region was voxelized into 125×10^6 cubic voxels (500, 500, 500). All the rays intersected by the walls of the cavity, including the secondary rays associated with the reflections in the interior of the cavity, were postprocessed in FluxTracer.

Figures 4(a) and 4(b) show the distribution of the radiant energy on a cross section along the x - z and y - z planes, respectively, showing how the radiant energy density is distributed within the cavity. This type of analysis can provide insights into the workings of the receiver and contribute to its optimization.

5.2 Point Analysis. In this analysis, given a user-defined point within the region of interest and a set of rays provided by the MCRT, for each one of the rays in the set, FluxTracer computes a distance and a three-dimensional point. The distance is the minimum distance from the ray to the user-defined point. The point is the location in space where this minimum distance takes place.

To carry out the point analysis, the user should specify to FluxTracer the following information:

- (1) The size and location of the region of three-dimensional space of interest, as in the radiant density analysis.
- (2) How the region of interest shall be voxelized, as in the radiant density analysis.
- (3) The point of interest regarding which the analysis should be carried out, specified by its coordinates (x , y , z).
- (4) The set of rays generated by the Monte Carlo ray tracer to be processed by FluxTracer.

Since the number of rays processed by FluxTracer is relatively large, it is not usually practical to provide for each ray its closest distance to the point of interest and the coordinates of the points of closest distance. Because of this, by default, FluxTracer provides instead the probability distribution of distances of the rays to the point of interest and the density distribution of these points

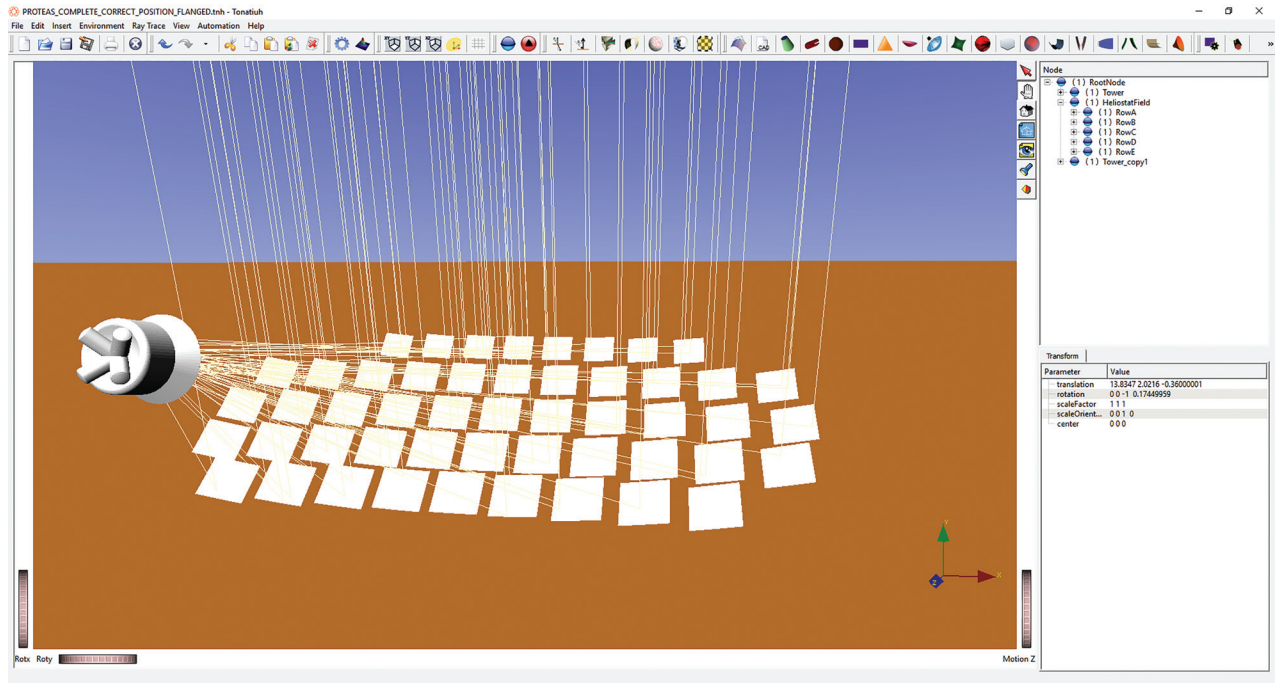


Fig. 3 The iStore cavity receiver and the CSP–DSW field facility being modeled and simulated in Tonatiuh

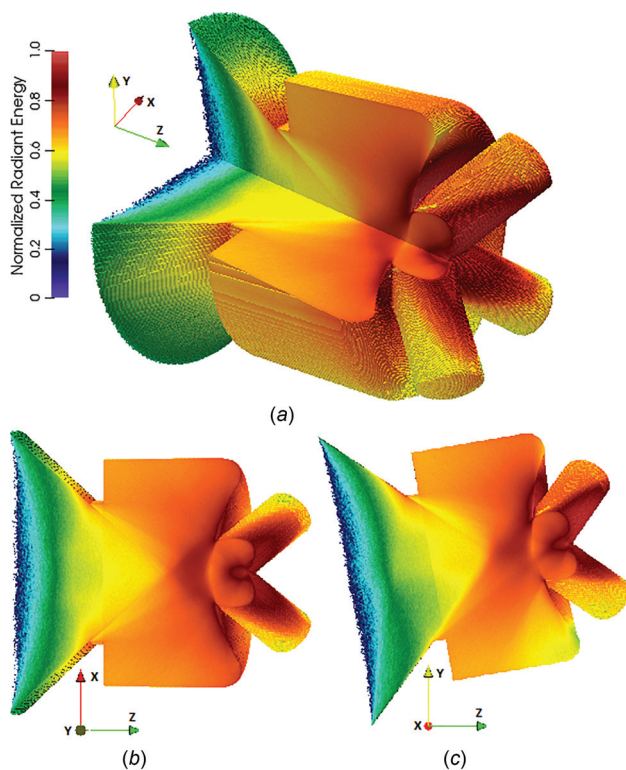


Fig. 4 Relative distribution of solar radiant density within the cavity of the Cyprus Institute's iStore receiver at noon summer solstice: (a) quarter cross section, (b) x - z horizontal plane cross section, and (c) y - z vertical plane cross section

in three-dimensional space using the same voxelization of the region of interest than in the radiant energy analysis.

Since the point analysis provides information regarding the minimum distances of the rays to a point of interest defined by the user, this type of analysis can be used to get an idea of the minimum region that should be analyzed around the point of interest

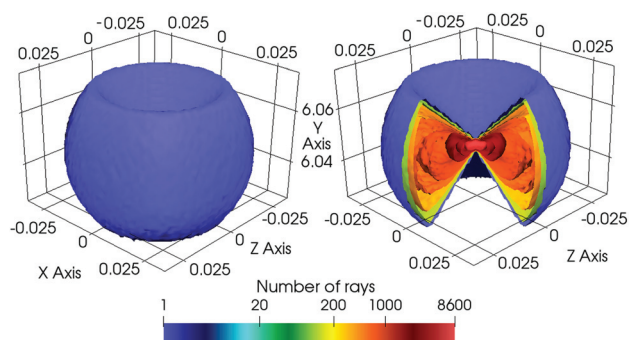


Fig. 5 Minimum convex receiver geometry to intercept all rays reflected from a parabolic dish of 10 m diameter and a focal length of 6.045 m assuming a pillbox sunshape of 4.65 mrad half-angle

to get insight on the distribution of radiant energy around that point. If the point of interest defined by the user is the focal point of the solar concentrator being analyzed, this type of analysis could provide insight on how to define a sensible region for the radiant density analysis or insight on what should be the dimensions and shape of a receiver to capture a given fraction of all the solar energy sent by the solar concentrator to the receiver. It could also facilitate the resolution of specific receiver related optimization problems without the need to run several times the Monte Carlo ray tracer or of generating new sets of rays.

Figure 5 shows the results obtained by carrying out with Flux-Tracer for the parabolic dish the point analysis described in Sec. 5.2. It provides the geometry of the smaller convex receiver that will intersect all the rays reflected by the dish, assuming a pillbox sunshape distribution of 4.65 mrad half-angle.

5.3 Line Analysis. This analysis is similar to point analysis, but with the user-defined line playing the role of the user-defined point. Thus, here given a user-defined line within the region of interest and a set of rays provided by the MCRT, for each one of the rays in the set, FluxTracer computes the minimum distance

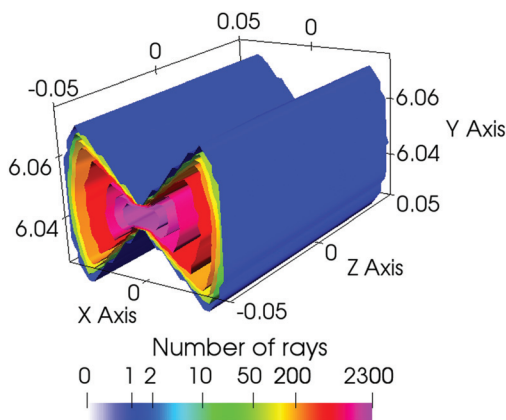


Fig. 6 Schematic showing a portion of the minimum convex receiver geometry to intercept all rays reflected from a parabolic trough of 10 m input aperture cross section and a focal length of 6.045 m assuming a pillbox sunshape of 4.65 mrad half-angle

from the ray to the user-defined line and the location in space where this minimum distance takes place.

To carry out the line analysis, the user should provide to FluxTracer the following information:

- (1) The size and location of the region of three-dimensional space of interest, as in the radiant density and point analysis.
- (2) How the region of interest shall be voxelized, as in the radiant density and point analysis.
- (3) The line of interest regarding which the analysis should be carried out, specified by point of coordinates (x, y, z) and direction in terms of a unit vector (d_x, d_y, d_z) .
- (4) The set of rays generated by the Monte Carlo ray tracer to be processed by FluxTracer.

As in the point analysis, because the number of rays usually processed by FluxTracer is relatively large, it is not usually practical to provide for each ray its closest distance to the line of interest and the coordinates of the points of closest distance. Here again, by default, FluxTracer provides instead the probability distribution of distances of the rays to the line of interest and the density distribution of this points in three-dimensional space using a similar voxelization of the region of interest than in the radiant energy analysis.

Since the line analysis provides information regarding the minimum distances of the rays to a user-defined line of interest, this type of analysis can be used to get an idea of the minimum region that should be analyzed around the line of interest to get insight on the distribution of radiant energy around that line. If the line of interest defined by the user contains the focal point of the solar concentrator being analyzed, this type of analysis could provide insight on what should be the dimensions and cross section of a cylindrical receiver to capture a given fraction of all the solar energy sent by the solar concentrator to the receiver. As in the case of point analysis, this type of analysis could also facilitate the resolution of specific receiver related optimization problems without the need to run again the Monte Carlo ray tracer or of generating new sets of rays.

Figure 6 shows the results obtained by carrying out a line analysis with FluxTracer for the parabolic trough presented in Sec. 5.1. It provides the geometry of the minimum convex receiver that will intersect all the rays reflected by the trough, assuming a pillbox sun-shape distribution of 4.65 mrad half-angle.

6 Preliminary Validation

A preliminary validation of FluxTracer was performed by analyzing with it a problem that has an analytical solution and

comparing the results provided by FluxTracer with the exact analytical solution.

Thirty-four years ago, Prof. Carlos Gomez-Camacho, from the University of Seville, Spain, derived the equation of the cross section of the minimum convex receiver tube that will intersect all sunrays reflected by an ideal parabolic trough if the sunshape is a pillbox of a given half-angle θ_s [27]. He identified four different equations for the cross section, which apply to four different cases determined by the relative position of the focal line of the parabolic trough with respect to the plane of the input aperture of the parabolic trough reflector. For this preliminary validation, only one of the four cases has been considered. This configuration corresponds to a parabolic trough whose focal line is above the level of input aperture of the parabolic trough reflector, which is the same parabolic trough configuration presented in Fig. 1(a) of Sec. 5.1. According to the analysis of Prof. Gomez-Camacho, for this configuration, the optimal receiver shape in polar coordinates is given by

$$r = \frac{2f \sin(\theta_s)}{1 - \cos\left[\phi \pm \left(\frac{\pi}{2} - \theta_s\right)\right]} \quad (1)$$

where r is the distance from the focal point, f is the focal length, θ_s is the half angle subtended by the sun, ϕ is the deviation angle from the horizontal line passing from the focal point, defined in the range $[\phi_i, 2\pi - \phi_i]$, and ϕ_i is the root of the following equation:

$$\frac{2f}{1 - \cos(\phi_i)} \sin(\phi_i) = W/2, \quad 0 \leq \phi_i \leq \pi/2 \quad (2)$$

where W is the length of the input aperture in the direction of parabola's cross section.

The above equation can be particularized for any given length of the input aperture cross section of the parabolic trough reflector and any given focal length. To compare with FluxTracer, the parabolic trough geometry was particularized for a length of the input aperture cross section of the parabolic trough reflector of 10 m and a focal length of 6.045 m. This geometry was modeled and simulated with Tonatiuh with a "scene" like the one presented in Fig. 1(a) of Sec. 5.1, which is composed of a parabolic trough reflector and a virtual plane parallel to the horizontal $x-z$ plane, located 8 m above the focal line. For the ray-tracing simulations with Tonatiuh, the sunshape

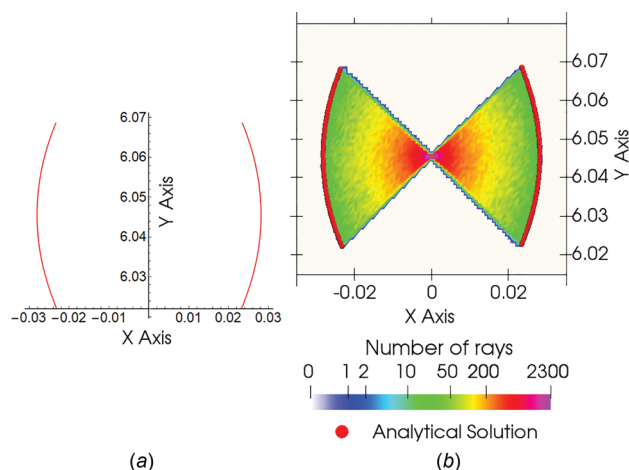


Fig. 7 Preliminary validation of FluxTracer. (a) Analytically calculated boundaries of the cross section of the minimum convex receiver geometry to intercept all rays reflected from a parabolic trough of 10 m input aperture cross section and a focal length of 6.045 m assuming a pillbox sunshape of 4.65 mrad half-angle. (b) Cross section of the density distribution in three-dimensional space of the point cloud resulting from FluxTracer's line analysis of the same parabolic trough.

was modeled as a pillbox with half acceptance (θ_s) of 4.65 mrad; the solar direct normal irradiance was set to 1 kW/m²; and for simplicity, the sun was assumed to be on the zenith and in the plane of symmetry of the parabolic trough reflector.

The set of rays generated by Tonatiuh was input to FluxTracer and used to carry out a line analysis. A region of interest of 0.12 m × 0.12 m × 8 m was chosen, centered around the focal line of the parabolic trough reflector. A cross section of this region of interest on a plane perpendicular to the focal line of the parabolic trough provides an approximation to the analytical cross section derived by Prof. Gomez-Camacho. As Fig. 7 shows, this approximation is indeed very accurate.

To estimate the accuracy of the FluxTracer cross section against the analytical solution, the mean percentage error can be used. It is defined as follows:

$$\text{MPE} = \frac{1}{N} \sum_{k=1}^N \left(\frac{X_k^e - X_k^a}{X_k^a} \right) \times 100\% \quad (3)$$

where the upper-script “a” refers to the analytical points, the upper-script “e” refers to the estimates provided by FluxTracer, and N is the total number of the outermost voxels/sampling points, as shown in Fig. 7(b)

As expected, the mean percentage error obtained was relatively low—less than 1%.

7 Conclusions

The main conclusions, which can be drawn from the work presented in this paper, are the following:

- (1) FluxTracer is the first of a series of software tools, which are part of an open source software framework CyI is developing, in collaboration with ANU to assist in the design and optimization of CST systems.
- (2) This framework is engineered from the start to be able to benefit from the use of HPC clusters and will be integrated for a variety of tools.
- (3) This paper presents the first one of those software tools, which is called FluxTracer.
- (4) The main purpose of FluxTracer is to provide the user with the possibility of carrying out diverse types of analysis of the solar radiation on a region of interest within a solar concentrator, such as a parabolic through, a parabolic dish, or a heliostat field.
- (5) These analyses are carried out by postprocessing a large set of rays generated by modeling and simulating the optical behavior of the solar concentrator of interest with a Monte Carlo ray tracer, such as Tonatiuh.
- (6) Currently, the main functionalities of FluxTracer are as follows:
 - (a) *Radiant density analysis*: In this analysis, FluxTracer computes the density distribution of radiant energy within the region of interest during a specified time interval.
 - (b) *Point analysis*: In this analysis, FluxTracer computes the minimum distance from each ray, within the large set of rays generated by the Monte Carlo ray tracer, to a user-defined point within the region of interest. It provides information regarding the minimum distances of the rays to the point and the position of the points on the rays corresponding to those minimum distances.
 - (c) *Line analysis*: In this analysis, FluxTracer computes the minimum distance from each ray, within the large set of rays generated by the Monte Carlo ray tracer, to a user-defined line within the region of interest. It provides information regarding the minimum distances of the rays to the line, and the position of the points on the rays corresponding to those minimum distances.

- (7) The usefulness of such series of analysis has been demonstrated.
- (8) A preliminary validation of FluxTracer has also been successfully carried out.

8 Future Work

After the successful development of the proof-of-concept, the next steps in the development of FluxTracer are to

- evolve rapidly from the development of a proof-of-concept to the development of the first beta release of the program;
- test the scalability of the software being developed;
- set up a complete development environment and make the program publicly available to facilitate third party contributions to the development of the program.

Acknowledgment

This work was partially supported by the European Union’s Horizon 2020 Research and Innovation Programme within the context of the CySTEM ERA Chair project, under grant agreement no. 667942, and as part of ASTRI, a project supported by the Australian Government, through the Australian Renewable Energy Agency (ARENA).

Funding Data

- Australian Renewable Energy Agency (Australian Solar Thermal Research Initiative, Funder ID. 10.13039/501100005105).
- Horizon 2020 Framework Programme (Cyprus Solar Thermal Energy Chair for the Eastern Mediterranean—CySTEM/667942, Funder ID. 10.13039/100010661).

Nomenclature

ANU	= Australian National University
CyI	= The Cyprus Institute
CySTEM	= CyI Solar Thermal Energy Chair for the Eastern Mediterranean
ERA	= European research area
f	= focal length
HPC	= high performance computing
LCC	= light collection and concentration
MCRT	= Monte Carlo ray tracer
N	= total number of outermost voxels/sampling points
r	= distance from an origin
W	= length of the input aperture in the direction of parabola’s cross section
θ_s	= half-angle subtended by the Sun
ϕ	= deviation angle from the horizontal line

References

- [1] IEA, 2015, “Technology Roadmap Solar Thermal Electricity,” *IEA Technology Roadmaps*, OECD Publishing, Paris, France.
- [2] Mathiesen, P., and Kleissl, J., 2013, “Using High-Resolution NWP to Forecast Solar Irradiance for the Coastal Regions of Southern California,” AMS General Meeting, Austin, TX, Jan. 9.
- [3] Cruz, N. C., Redondo, J. L., Berenguel, M., Álvarez, J. D., Becerra-Teron, A., and Ortigosa, P. M., 2017, “High Performance Computing for the Heliostat Field Layout Evaluation,” *J. Supercomput.*, **73**(1), pp. 259–276.
- [4] He, C., Feng, J., and Zhao, Y., 2017, “Fast Flux Density Distribution Simulation of Central Receiver System on GPU,” *Sol. Energy*, **144**(1), pp. 424–435.
- [5] Chiesi, M., Vanzolini, L., Scarselli, E. F., and Guerrieri, R., 2013, “Accurate Optical Model for Design and Analysis of Solar Fields Based on Heterogeneous Multicore Systems,” *Renew. Energy*, **55**, pp. 241–251.
- [6] Cho, E., and Miller, F., 2016, “Comparing the Monte Carlo Method to the Discrete Ordinates Methods in Fluent for Calculating Radiation Heat Transfer in a Small Particle Receiver,” *ASME Paper No. ES2016-59465*.

- [7] Mecit, A. M., and Miller, F., 2014, "A Comparison Between the Monte Carlo Ray Trace and the Fluent Discrete Ordinates Method for Treating Solar Input to a Particle Receiver," *ASME Paper No. ES2014-6412*.
- [8] Belhomme, B., Pitz-Paal, R., Schwarzbözl, P., and Ulmer, S., 2009, "A New Fast Ray Tracing Tool for High-Precision Simulation of Heliostat Fields," *ASME J. Sol. Energy Eng.*, **131**(3), p. 031002.
- [9] Farges, O., Béziau, J., Bru, H., Hafi, M., Fournier, R., and Spiesser, C., 2015, "Life-Time Integration Using Monte Carlo Methods When Optimizing the Design of Concentrated Solar Power Plants," *J. Sol. Energy*, **113**, pp. 57–62.
- [10] Ghoheity, A., and Mitsos, A., 2012, "Optimal Design and Operation of a Solar Energy Receiver and Storage," *ASME J. Sol. Energy Eng.*, **134**(3), p. 031005.
- [11] Craig, K. J., Moghimi, M. A., Rungasamy, A. E., Marsberg, J., and Meyer, J. P., 2016, "Finite-Volume Ray Tracing Using Computational Fluid Dynamics in Linear Focus CSP Applications," *Appl. Energy*, **183**, pp. 241–256.
- [12] Moghimi, M. A., Craig, K. J., and Meyer, J. P., 2015, "A Novel Computational Approach to Combine the Optical and Thermal Modelling of Linear Fresnel Collectors Using the Finite Volume Method," *Sol. Energy*, **116**, pp. 407–427.
- [13] Scott, P., De la Calle, A., Hinkley, J., and Pye, J., 2017, "SolarTherm: A Flexible Modelica-Based Simulator for CSP Systems," *AIP Conf. Proc.*, **1850**, p. 160026.
- [14] Blanco, M. J., Amieva, J. M., and Mancillas, A., 2005, "The Tonatiuh Software Development Project: An Open Source Approach to the Simulation of Solar Concentrating Systems," *ASME Paper No. IMECE2005-81859*.
- [15] Blair, N., Dobos, A. P., Freeman, J., Neises, T., Wagner, M., Ferguson, T., Gilman, P., and Janzou, S., 2014, "System Advisor Model, SAM 2014.1.14: General Description," National Renewable Energy Laboratory, Golden, CO, Technical Report No. [NREL/TP-6A20-61019](https://www.nrel.gov/pdfs/TP-6A20-61019.pdf).
- [16] Pye, J., Abbasi, E., Arjomandi, M., Coventry, J., Ghanadi, F., Hughes, G., Kilm, J. S., Ma, L., Shirazi, A., Torres, J. F., Venn, Wang, Y., and Zheng, M., 2018, "Towards Testing of a Second-Generation Bladed Receiver," *SolarPACES International Conference, Casablanca, Morocco, Oct. 2–5*.
- [17] Blanco, M., Mutuberria, A., Monreal, A., and Albert, R., 2011, "Results of the Empirical Validation of Tonatiuh at Mini-Pegase CNRS-PROMES Facility," *SolarPACES International Conference, Granada, Spain, Sept. 20–23*.
- [18] Born, B., and Wolf, E., 1980, *Principles of Optics*, 6th ed., Pergamon, Oxford, UK, p. ii.
- [19] Maurice, H., and Nir, S., 2012, *The Art of Multiprocessor Programming*, Elsevier, Amsterdam, The Netherlands, p. 14.
- [20] Amanatides, J., and Woo, A., 1987, "A Fast Voxel Traversal Algorithm for Ray Tracing," *European Computer Graphics Conference and Exhibition*, Amsterdam, The Netherlands, Aug. 24–28, pp. 3–10.
- [21] Bonanos, A. M., Georgiou, M. C., Guillen, E., and Papanicolas, C. N., 2017, "CSP+D: The Case Study at the PROTEAS Facility," *AIP Conf. Proc.*, **1850**, p. 170001.
- [22] Stokos, K. G., Stiliaris, E., Bonanos, A. M., Georgiou, M. C., Guillen, E., Montonen, A., and Papanicolas, C. N., 2017, "The Control System at Proteas," *SolarPACES International Conference, AIP Conf. Proc.*, **2033**, p. 210019.
- [23] Bonanos, A. M., and Papanicolas, C. N., 2016, "The PROTEAS Facility Heliostats Field: Layout and Performance Prediction," *Fifth International Conference on Renewable Energy Sources and Energy Efficiency, Nicosia, Cyprus, May 5–6*.
- [24] Votyakov, E. V., and Papanicolas, C. N., 2016, "Consistent Multiphysics Simulation of a Central Tower CSP Plant as Applied to ISTORE," *AIP Conf. Proc.*, **1850**, p. 160027.
- [25] Votyakov, E. V., Georgiou, M. C., Guillen, E., Stiliaris, E., and Papanicolas, C. N., 2017, "Experimental Methodology to Calculate Thermal Losses of a Molten Salt Cavity Receiver," *AIP Conf. Proc.*, **2033**, p. 040041.
- [26] Papanicolas, C. N., Bonanos, A. M., Georgiou, M. C., Guillen, E., Jarraud, N., Marakkos, C., Montonen, A., Stiliaris, E., Tsioli, E., Tzamtzis, G., and Votyakov, E. V., 2015, "CSP Cogeneration of Electricity and Desalinated Water at the Pentakomo Field Facility," *AIP Conf. Proc.*, **1734**, p. 100008.
- [27] Camacho, G. C., 1984, "Thermodynamic Criteria for Optimizing Receiver Shaping and Rim Angle," *SSPS DCS International Workshop: The First Term, Cologne, Germany, pp. 2.4/1–2.4/52*.