SASEC
1st Southern African Solar Energy Conference
21-23 May 2012

REVIEW OF OPTICAL SOFTWARE FOR USE IN CONCENTRATING SOLAR POWER SYSTEMS

Sebastian-James Bode[1] **and Paul Gauché**[2]

[1]B.Sc (Mech) Masters student, University of Stellenbosch, Private Bag X1, Matieland Stellenbosch, 7602, South Africa, Phone: +27-74-1309-442, E-Mail:14922126@sun.ac.za

[2]M.Sc. Eng. (Mech), Sr. Researcher & Coordinator – Solar Thermal Energy Research Group

Abstract

A review of the main software tools used in the optical design, analysis and optimisation of central receivers systems has been performed. The review was performed using all the resources available to a post graduate researcher. The functionality of the codes was explored and it was found that the codes could be divided into two broad categories: those used to analyse and optimise field layout and those capable of accurately simulating the flux incident on the receiver from one or more heliostats. As of yet, no code reviewed can perform both tasks. Emphasis was placed on the availability of the codes. It was found the Biomimetic and CRS4-2 codes were unavailable, HFLD and TieSOL can be purchased commercially/academically, SPRAY, STRAL and HFLCAL can be purchased commercially or possibly obtained through collaboration, SolTrace is free to download and use and Tonatiuh is open source. The possibility of writing one's own code was also explored.

Keywords: Ray tracing, concentrating solar power software, solar flux calculation, optical modeling.

1. Introduction

In central receiver systems (CRS), incoming solar radiation is reflected by a multitude of mirrors, or heliostats, and concentrated in the aperture of a receiver to be transformed into another form of energy. From an optical point of view, the precise prediction of flux density distribution produced by a heliostat or heliostat field on the receiver aperture is indispensable for a CRS researcher. For an industrial project, prediction of the power incident on the receiver gives a valuable indication of a CRS economic feasibility. This paper serves as a guide of the main optical software tools used to model central receiver system.

This is not the first review of its kind. Most notably a comparative overview by performed by Garcia et al. in 2007 [1]. However, since 2007 almost all the codes reviewed by Garcia et al. have either been upgraded or are no longer available. Furthermore, with a renewed interest in CRS several new tools have been developed.

This review differs slightly from that of previous work as it is conducted to answer the question of what tools are available (and appropriate) for use in a South African solar energy research institute. The main criteria addressed were code functionality and availability (which codes are academically or otherwise available, which codes are propriety licensed and which codes no longer exists).

The following section provides an overview of the different codes, their capabilities, calculation methods and availability. Section 3 provides a summary of the codes as well as references on where one can obtain the codes. Section 4 provides some insight into the codes and code selection followed by conclusions.

2. Codes

Design analysis and simulation software for CRS have been a critical component of the technology since its inception. The first codes developed for CRS came from studies carried out by the US for Solar One[1] in the late 70's [1]. A handful of codes were developed which could handle field layout optimization, receiver flux

[1]Solar One was a 10MW central receiver pilot project complete in 1982 in the Mojave Desert, USA [4]

simulation and overall plant performance prediction. Among these earlier codes is the heavily cited DELSOL.

DELSOL, or its windows adaption, winDELSOL is capable of predicting the optical performance of a heliostat field, optimizing field layout based on energy costs and unlike most of the optical tools reviewed has a economic model of central receiver components [2]. The cost models in winDELSOL were developed from actual and estimated costs from conceptual CRS systems. As winDelsol has been so frequently referenced considerable effort was put into obtaining a copy of the code. However, with the best efforts and having exhausted all resources, a copy of DELSOL software could not be obtained.

Similarly, the first generation code HELIOS, which uses cone optics to calculate flux density from heliostat fields, could not be obtained. Although HELIOS is still used at SANDIA, it is unsupported and difficult to use. It was suggested that a more modern user friendly software be used for the same function [3].

The first generation codes were constrained by the computing power at the time. Therefore attempts were made to minimize computing resources. With modern multi-core computer systems there has been paradigm change in software development toward parallel and multithread programming. That is not to say that the first generation codes are obsolete - the algorithms and routines are still applicable today. Thus, a few of these early generation codes namely MIRVAL, UHC and HFCAL have been upgraded.

2.1. SPRAY (MIRVAL)

MIRVAL, developed by SANDIA labs, was one of the first Monte Carlo ray tracing programs written for heliostat optical performance [4]. Monte Carlo ray tracing methodology consists of following stochastic paths of a large number of rays as they interact with the surfaces. Each ray has a specific direction and carries a certain amount of energy. The irradiance of a surface is proportional to the number of impacting rays, and the reflection of the rays depends on the emissive, reflective, and absorptive behaviour in the surface [5].

In its original form MIRVAL is no longer available but development was taken over by DLR where it forms the core of a FORTRAN based code SPRAY. SPRAY can be used to calculate field efficiencies as well as flux maps from individual heliostats or fixed heliostat fields. The main advantage of SPRAY is that it has a large number of features and geometries built in due to its long history of usage [6] (development of MIRVAL began in the 70's [7]). SPRAY does not have a user interface and is operated via ASCII files, which may prove to be difficult to start with but for many tasks it offers ready solutions [6].

SPRAY is commercially available through DLRs Institute of Solar Research and may also be made available through academic collaborations [6].

2.2. University of Houston field codes(TieSOL)

The University of Houston field codes (UHC) or RCELL codes are a suite of four codes which deal mainly with the optical design and optimization of heliostat fields and receivers [4]. The codes are no longer in a deliverable condition. They can only be run at the University of Houston on an outdated computer, using special code conversions. They are however being updated by the software company Tietronix, who are developing them into a commercial package known as TieSOL [8]. The release of Tiesol with field layout capabilities is unknown.

Although the UHC are no longer available an original author Lorin Vant-Hull has documented the concepts and procedures in several publications [8].

2.3. HFLCAL

Development of HFLCAL started in the early 80's by Michael Kiera and was developed for two main tasks; the calculation of the annual plant output at a given configuration and the layout and optimization of a total system. Today it continues to be used and developed by the DLR, who use it for the layout and optimization of heliostat fields. The software uses a simplified mathematical model of concentrator optics, modelling the reflected image of each heliostat by a circular normal distribution. Although ray tracing techniques have the advantage of reproducing real interactions between reflective surfaces [9], each ray has to be modelled, which requires large computation times compared to simpler mathematical models. Few of the codes reviewed employed Monte Carlo methods for field optimization. HFLCAL features include: automatic multi-aiming, secondary concentrator optics, tower reflector systems, various receiver models and the ability of least-cost optimization [10].

HFLCAL is commercially available through DLR as well as through possible collaborations [6].

2.4 SolTrace

SolTrace is a Monte Carlo ray tracing program which, in 2011, was completely rewritten from the ground up. The latest C++ version uses parallel processing techniques which, in theory a computer with *n* processor will experience speed *n*x over single processors. SolTrace can model and characterise the optics of a single heliostat or with the aid of a built in scripting language, be used to model large optical systems [9]. Several geometric heliostat and receiver shapes are available. SolTrace can model and output flux maps for each stage. SolTrace has a Google SketchUp plug-in for visualization.

SolTrace is freely available and can be downloaded from the NREL website [11].

2.5. Tonatiuh

Tonatiuh is an open source Monte Carlo ray tracing program designed for the analysis and simulation of the optics and energy behaviour of CSP systems. Similar to SolTrace, Tonatiuh has several geometric heliostat and receiver shapes and multiple stages can be modelled. Tonatiuh has an inbuilt visualizer. However, flux maps need to be generated with an external program such as Python or Mathematica. Tonatiuh has a GNU General Public License which allows free access to the source code for anyone interested in using or contributing to its development. Access to the source code allows a user to develop Tonatiuh to suit almost any application or requirement. Tonatiuh has been experimentally validated and the results presented at the 2011 SolarPACES [12].

Tonatiuh is free to download from its website [13].

2.6. STRAL

STRAL is a completely new ray tracing software which generates rays on the surface of the heliostat, as opposed to generating the rays in a plane above the heliostats. As no rays ever miss the target, it is computationally more efficient than other tools. STRAL enables the setup of heliostat field models in great detail using highly resolved heliostat mirror surface and geometry data as well as real sun shapes and blocking and shading [14].

STRACL is commercially available through DLR as well as through possible collaborations [6].

2.7. TieSol

The TieSol suite uses the parallel processing power of Graphic Processing Units (GPU) to implement extremely fast Monte Carlo ray tracing, well beyond the currently available capabilities of other software. The software suite allows for the design, analysis and optimization of CRS systems. This is achieved by analysing the effects of different optical and mechanical errors on the field, reciever flux map computation, as well as efficiency and annual performance computation [15]. Tietronix has developed an advanced visulisation tool for TieSol capable of displaying the heliostat tracking in real time [16].

TieSOL is commercially available from Tietronix and is sold in individual modules [17].

2.8. ISOS

ISOS was developed to improve the durability of receivers. The MATLAB based program uses a numerical algorithm to calculate regions of homogenous flux (isosurfaces). A 3D flux map is generated that allows the user to assess the flux at any height above the focusing heliostat [18].

The code is available for academic use, but requires input data from a separate ray tracing program [19].

2.9. Heliostat Field Layout Design (HFLD)

HFLD is a MATLAB code for field layout design based on the edge ray principal of non-imaging optics. The edge ray principal simply states that if the limiting rays (rays coming from the edges of the source) are transferred to the receiver, this will ensure that all rays coming from the inner points in the source will end up on the receiver [20]. The accuracy and feasibility of the HFLD code has been confirmed by comparing with published data from the PS10[2] field. When compared with other codes, such as winDELSOL, HFLD has a shorter computational time during design and optimization of the heliostat field. The code also calculates the annual sunshine and duration on the land surface between heliostats, to evaluate the feasibility of crop growth [21].

The HFLD code can be purchased for commercial or academic use [22].

2.10. CRS4-2

CRS4-2 is a FORTRAN based code used for the simulation of optical performance of CRS systems. For an arbitrarily arranged field it can calculate cosine, shading and blocking effect through tessellation of the heliostats [23].

At the moment CRS4-2 is not readily available but discussions are underway to allow sharing with external institutes or setting up collaborations [24]

2.11. 'Biomimetic'

An altogether different approach uses a biomimetic pattern for heliostat field layout optimization. Biomimicry is the emulation of nature in man-made structures. The tool is able to perform calculation of annual average optical efficiencies accounting for cosine losses, shading and blocking, aberration as well as atmospheric attenuation. The PS10 plant was used as a demonstration application; the model showed an improvement on existing configurations of 0.36% in efficiency with a 15.8% reduction in land usage, thus minimizing the levelized cost of electricity [25].

At the moment the code is under pending patent and is unavailable but the developers are willing to licence the code under conditions decided by the MIT Technology Licensing Office [26].

2.12. Other Codes

One of the codes investigated by Garcia et al. [1] was Fiat Lux. Fiat Lux was initially designed to validate the optical quality of heliostats. Fiat Lux could not be obtained, nor could reference to if be found in any recent open literature. Similarly, no recent publication could be found on the codes OPTEC, SOLVER, SENSOL or SCT mentioned by Garcia et al. [1], nor could the authors of these codes be reached. These codes were therefore excluded from this review.

Although performed as comprehensively as possible, this review is not exhaustive. Proprietary industrial ray tracers were not explored, nor were graphical ray tracing software (which in theory can also be utilized).

3. Summary

The following two tables provide a summary of all the software codes reviewed. Special attention was paid to

[2] PS10 (Spanish: Planta Solar 10) is the worlds first comercial CR plant. The 11MW plant is located in Andalucia, Spain.

the availability of each code for an academic researcher; therefore Table 1 includes the references to the authors of the codes so that interested parties can contact them to obtain a copy. Table 2 provides a visual representation of the functions of the various codes.

	Organization	Function	Calculation Method	Availability	Ref.
Spray	DLR	Heliostat and Field simulation	Monte Carlo ray tracing	Commercially available	[10]
SolTrace	NREL	Heliostat and Field simulation	Monte Carlo ray tracing	Free to use	[11]
Tonatiuh	CENER	Heliostat and Field simulation	Monte Carlo ray tracing	Open source	[13]
STRAL	DLR	Heliostat and Field simulation	Backward ray tracing	Commercially available	[14]
Tiesol	Tietronix	Heliostat and Field simulation	Backward ray tracing	Commercially available	[15]
ISOS	Universidad Nacional Autónoma de México	Flux distribution of individual heliostat	Mathematical algorithm	Academically available	[18]
HFLCAL	DLR	Layout and optimization of heliostat fields	Normal distribution	Commercially available	[10]
CRS4	CRS4	Simulation of CRS plants	Tessellation	Not at present	-
HFLD	Chinese Academy of Science	Design and analysis of CRS field layout	Edge ray	Commercially available	[21]
Biomem.	MIT	Design and analysis of CRS field layout	Biometry	Pending	-

Table 1:Summary of the main features of the codes, and code availability

	Heliostat quality	Flux due to single heliostat	Flux map due to field	Field simulation (S&B)	Instant. Th. power on receiver	Yearly efficiency	Field layout optimiza-tion
SPRAY							
SolTrace							
Tonatiuh							
STRAL							
Tiesol							
ISOS							
HFLCAL							
CRS4							
HFLD							
Biomem.							

Table 2: Summary of code capabilities

As can be seen from Table 2, the TieSOL suite has the greatest functionality of the all the codes (more so when the field optimization tool is released). ISOS is capable of calculating the instantaneous flux on a receiver, however only from one heliostat. The HFLCAL code can produce a flux map, however with limited accuracy. Theoretically, SPRAY, SolTrace, Tonatiuh and STRAL can all perform the same task and differ only in user-friendliness, computational time and accuracy. Without a user interface SPRAY can, admittedly, be difficult to get used to. STRAL boasts a technique for computationally fast ray tracing and preliminary validation of Tonatiuh compared with SolTrace showed similar results between the two codes [27]. However, without a side by side comparison of all the codes no definite conclusions can be drawn and care should be taking when selecting a code.

4. Code Selection

Garcia et al. [1] suggested a strategy of code selection for industrial projects, to first determine the general layout of the plant in terms of tower height, heliostat position etc., then perform a detailed study including a closer description of the heliostat flux and field performance. This latter task can be performed with any of the Monte Carlo ray tracers, while the layout can be performed with the HFLCAL or HFLD codes. The authors then suggest that the system is modelled with tools such as TRYNSYS, which can model the transient behaviour of thermal systems [28], or the Solar Advisor Model (SAM), which supports industry calculations of the cost of energy [29]. For a researcher, there is no standard tool.

This review was performed from the perspective of a researcher in South Africa. By its very nature, research explores different and novel ideas. Each one of the tools reviewed had their own strengths but equally, they come with their own limitations. For research one wants a tool with the capability and flexibility to model their particular system or application. It is suggested therefore that the option of developing and writing one's own tool should not be overlooked. A CRS researcher need not start from scratch. Several papers have been published in which the mathematical models and algorithms have been described in detail to allow for readers to develop their own code. However an overview of all these papers is beyond the scope of this review.

For the solar research laboratory at the University of Stellenbosch it has been the authors' decision to develop a new ray-tracing code for solar flux calculations. For the authors particular area of research it was found that only a few commercial codes were applicable. Of the applicable codes all were unduly expensive for the research institute. The free software packages SolTrace and Tonatiuh were therefore critically investigated.

SolTrace is downloaded as an executable file and the inner workings of the software are unknown to the user. Without knowledge of the strengths and limitations of the modelling methods it is difficult to identify causes of discrepancies between experimental and simulated results. Tonatiuh bridges this gap by providing source code of the software. Unfortunately, it was found to be very poorly documented and commented. It was decided therefore to write a new code specifically designed for the authors' research, but with the framework for future development.

By developing a code from grass roots and providing a framework for future development there is the opportunity to build the capacity in understandings of the optics in solar research in South Africa. The exact procedures and algorithms of the code are extensively documented which will aid in future development. A well-documented code also allows for the software to be shared setting up broader collaborations among solar research institutes.

5. Conclusion

A review of the main publically or commercially available software tools used for CRS design and analysis was performed. The codes can be divided among two categories namely codes used for detailed plant performance simulation and codes used for optimal field layout. All the codes differ slightly from one another when it comes to calculation techniques or optimization parameters which make selecting the ideal code for research a challenging task. Of the codes used for plant performance only SolTrace and Tonatiuh are free and readily available to CRS researcher.

The option of developing one's own code should not be disregarded as none of the tools reviewed were developed in South Africa. Writing a new and unique tool may lay the foundation to help build the capacity of solar research in South Africa, as well as set up the groundwork for possible inter/intra institutional collaboration.

Acknowledgements

The authors acknowledge the financial support from STERG provided by DST and NRF Solar Thermal Spoke as well as Stellenbosch University Hope Project. The authors wish to thank Thomas Roos and Derek Griffith at the CSIR and Clifford Ho form Sandia Labs for their invaluable advice.

References

[1] P. Garcia, A. Ferriere, J-J Bezian, "Codes for solar flux calculations dedicated to central receiver system applications: A comparative review," *Solar Energy*, vol. 82, pp. 189-197, 2008.

[2] B. L. Kistler, "A user's manual for Delsol3: A computer code for calculating the optical performance and optimal system design for solar thermal central receiver plants," SANDIA, SAND86-8018, 1986.

[3] C. Ho, NREL, 2012, Personal Communication.

[4] P. Falcone, "A Handbook for Solar Central Receiver Design," Sandia Laboratories, Livermore, SAND 86-8009, 1986.

[5] X.L. Xia, H. P. Tan Y. Shuai, "Radiation performance of dish solar/cavity receiver systems," *Solar Energy*, vol. 82, pp. 13-21, 2008.

[6] Peter Schwarzbözl, DLR - German Aerospace Center, 2012, Personal Communication.

[7] J. D. Hankins, P. L. Leary, "User's Guide for Mirval - A computer code for modeling the optical behavior of reflecting solar concentrators," SANDIA, Livermore, SAND77-8280, 1979.

[8] L. Vant-Hull, University of Houston, 2011, Personal communication.

[9] NREL, SolTrace 2011.7.5 Help File, 2011, Available: In SolTrace. http://www.nrel.gov/csp/soltrace/download.html

[10] M. Schmitz, R. Pitz-Paal, P. Schwarzbözl, "Visual HFLCAL - A software tool for layout and optimization of heliostat fields," in *SolarPACES*, Berlin, 2009.

[11] National Renewable Energy Laboratory. (2011, July) SolTrace: Downloads. [Online]. http://www.nrel.gov/csp/soltrace/download.html

[12] A. Mutuberria, A. Monreal, A. Albert, M. Blanco, "Results of the empirical validation of Tonatiuh at Mini-Pegase CNRS-PROMES facility," in *SolarPACES*, Seville, 2011.

[13] Tonatiuh. (2011) Tonatiuh. [Online]. http://code.google.com/p/tonatiuh/

[14] R. Pitz-Paal, P. Schwarzbözl, S. Ulmer, B. Belhomme, "A new fast ray tracing tool for high-precision simulation of heliostat fields," *Solar Energy Engineering*, vol. 131, August 2009.

[15] P. Armstrong, C. Nilsson, N. Vu, M. Izygon, "TieSOL – A GPU-Based Suite of Software for Central Receiver Solar Power Plants," in *SolarPACES*, Granada, 2011.

[16] M. Izygon, SolarPACES, 2011, Slide Presentation.

[17] M. Izygon, Tietronix, 2012, Personal Communication.

[18] M. Sánchez-González, C.A. Estrada, D. Riveros-Rosas, "Three-Dimensional analysis of a concentrated solar flux," *Journal of Solar Energy*, vol. 130, February 2008.

[19] D. Riveros, National University of Mexico, 2012, Personal communication.

[20] P. A. Davies, "Edge-ray principle of non-imaging optics," *Optical Society America*, vol. 11, no. 4, April 1994.

[21] Z. Lu, W. Yu, Z. Wang, X. Wei, "A new code for the design and analysis of the heliostat field layout for pwer tower system," *Solar Energyr*, vol. 84, pp. 685-690, 2010.

[22] Xiudong Wei, Changchun Institute of Optics, Fine Mechanics and Physics of Chinese Academy of Sciences, 2012, Personal Communication.

[23] D. D'Aguanno, E. Leonardi, "CRS4-2: A numerical code for calculation of the solar power collected in a central receiver," *Energy*, vol. 36, pp. 4828-4837, 2011.

[24] E. Leonardi, CRS4, 2012, Personal Communication.

[25] M. Torrion, A. Mitsos, C. J. Noone, "Heliostat field optimization: A new computationally efficient model and biomimetic layout," *Solar Energy*, vol. 86, pp. 792-803, December 2012.

[26] Alexander Mitsos, MIT, 2012, Personal Communication.

[27] M.J. Blanco, "Preliminary validation of Tonatiuh," in *SolarPACES*, Berlin, 2009.

[28] TRYNSYS. (2012) TRYNSYS. [Online]. www.trynsys.com

[29] NREL. (2012) NREL: Concentrating solar power research. [Online]. https://www.nrel.gov/analysis/sam/

[30] G. Zhu, M. J. Wagner, "A generic CSP performance model for NREL's System Advisor Model," in *SolarPACES*, Seville, 2011.

[31] C. Ho, "Software and Codes for Analysis of Concentrating Solar Power Technologies," SANDIA, SAND2008-8053, 2008.

[32] Tietronix. (2012, February) Tietronix. [Online]. http://www.tietronix.com/sites/default/files/Solar_DataSheet.pdf