SIMULATION OF YEARLY ENERGY FOR SOLAR HEATING SYSTEM
Olivier Farges, Jean-Jacques Bézian, Mouna El-Hafi, Hélène Bru

To cite this version:
Olivier Farges, Jean-Jacques Bézian, Mouna El-Hafi, Hélène Bru. SIMULATION OF YEARLY ENERGY FOR SOLAR HEATING SYSTEM. 18th SolarPACES Conference, Sep 2012, Marrakech, Morocco. hal-01165305

HAL Id: hal-01165305
https://hal-mines-albi.archives-ouvertes.fr/hal-01165305
Submitted on 18 Jun 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
SIMULATION OF YEARLY ENERGY FOR SOLAR HEATING SYSTEM

Olivier Farges1, 3, Jean-Jacques Bézian2, Moussa El Hafi2, and Hélène Bru3

1. M2S Student, Total New Energies and Centrale RAPSODIÈRE, UMR CNRS 5912, Campus Jardin, Route de Tollet, 81013 Albi Cedex 09, France
2. Centre RAPSODIÈRE, UMR CNRS 5912, École des Mines d’Albi-Carmaux, 81013 Albi Cedex 09, France
3. Total New Energies, R&D - Concentrated Solar Technologies, Immeuble Lafayette, 2 place des Vosges - La Défense 5, 92078 Paris La Défense Cedex, France

PROBLEM

The preliminary design step is one of the most important stage of the project process of a solar concentrating facility. In order to improve performances of the central receiver system (CRS), we developed a simulation tool based on Monte Carlo methods taking into account sun positions over the year to evaluate yearly energy at receiver in one simulation. With the efficient Monte Carlo Sun Tracking (MCST) algorithm, we obtain a fast and accurate code that permits to achieve optimization step in a reasonable time.

MODEL

The MCST algorithm evaluates the yearly energy at the entrance of a CRS receiver expressed by:

\[
A = \frac{1}{2\pi} \int_{0}^{2\pi} \int_{0}^{\pi} \sin \theta d\theta d\phi \times \frac{1}{2h} \int_{0}^{h} \text{DNI} \times [H(\theta, \phi, \tau)] \times \int_{H(\theta, \phi, \tau)}^{H(\theta, \phi, \tau')} \left(1\right)
\]

It also includes the reflection events involved in a CSP system and computes additional values characterizing CSP facility optical performances:

- Shadowing
- Blocking
- Spillage

As computational time is mainly devoted to update the heliostats orientation, we introduced an upgraded version with the Multi Rays Monte Carlo Sun Tracking (MRMCST) algorithm. For each dates, we trace 100 rays. As we obtain comparable results with less computational time, we know that it runs date by date. We evaluate as reference a "Tonatiuh equivalent" to MRMCST considering the following step execution time : 4.003s for each day, taking into account script opening (2s), pre and post-processing (2s) and tracing 100 rays (3ms).

INTERPOLATE DNI VALUES

We define each solar position with a day and a time of the day, a position on the Earth rotating around the axis, a day of the year, a latitude and a longitude. Each sun position corresponds to a DNI value according to weather pattern. Starting from hourly radiation datasets stored in database, we interpolate data with linear interpolation (ex. : 21/06 of June 2005 in Albi [1]).

ALGORITHM

Energy A = 0;

\textbf{foreach event do}

Uniforim sampling of \( \delta \) in [0, 365];

Uniform sampling of \( \eta \) in [7AM, 7PM];

for \( i = 1 \) to \( N^e \) do

Uniform sampling of \( x_1 \) on \( H^x \);  

Sampling of \( \theta_0 \) in solar disk \( \Omega_1 \);  

Generation of \( n_0 \) according to Blinn’s model;

if No shadowing between sun and \( x_1 \) then

\[
\begin{align*}
\theta_0 &= \text{DNI} \times \Omega_1 \times \eta \times \delta; \\
\theta_0 &= 0; \quad w_{\Omega_1} = 1; \quad \text{break};
\end{align*}
\]

Spacial reflection \( \Omega_1 = \Omega_2 + 2n_0 \times \Omega_5 \times \Omega_1 \);  

\( x_2 \) is intersection of \( \text{Ray}(x_1, \omega_1) \) with geometry element;

if \( x_2 \) exists then

if \( x_2 \in T \) then

\[
\begin{align*}
A &= A + w_A; \\
w_A &= 0; \quad w_\Omega = 1; \quad \text{break};
\end{align*}
\]

else

\[
\begin{align*}
A &= A + w_A; \\
w_A &= 0; \quad w_\Omega = 1; \quad \text{break};
\end{align*}
\]

\( i = i + 1 \);

EDSTAR

EDStar (numerical Environment of Development for Statistical Radiative simulation) is a coding environment [2]. Using Monte-Carlo methods, it takes advantages of advanced rendering techniques from computer graphics community. Three libraries are combined into the Mcm3D development environment:

- GNU Scientific Library (GSL) used for uniform random number sampling in the interval unit;
- Physically Based Rendering Techniques (PBRT) [4] provides a C++ object library to manage complex geometries;
- mcm C++ object library that handles Monte Carlo algorithms programming.

VALIDATION

We compare EDSTaR results with Tonatiuh [3] to compute a testing case: a tower and 146 heliostats (9 squared mirrors of 1.6 meter sided) in a heliostats field designed with the MUEEN [5] method, following a radial staggered layout. We make some general assumptions:

- Reflections are specular;
- CRS is located at the junction of the Greenwich meridian and the equator;
- The target is a square with 10m;

Firstly, we compute only one date at a time (4 dates tested at noon) to obtain a power value. Then, we randomly choose 50 dates and compute each date with Tonatiuh to approximate the average instantaneous value received by the receiver over one year (Number of rays = 10%). We only need to do one simulation with 50 rays to integrate over time and obtain an average instantaneous energy value. We see that EDSTaR gives an estimation in accordance with Tonatiuh results for a yearly simulation done date by date even if error bars are significantly large due to the small number of dates computed. By increasing the number of dates we obtain a more precise value of the yearly energy.

COMPUTATIONAL TIME

We compute computational time with Tonatiuh, knowing that it runs date by date. We evaluate as reference a “Tonatiuh equivalent” to MRMCST considering the following step execution time : 4.003s for each day, taking into account script opening (2s), pre and post-processing (2s) and tracing 100 rays (3ms).

CONCLUSION AND OUTLOOK

We present a new approach:

- To evaluate yearly energy at CRS receiver;
- Fast and accurate;
- Which can easily be integrated in an optimization process;

We plan to use it with typical year DNI data to design a solar field optimized on a yearly production basis.

REFERENCES