I. Troch, F. Breitenecker, eds.   ISBN 978-3-901608-35-3

Modelling and real-time simulation of heliostat fields in Central Receiver Plants

J. Bonilla[1], L. Roca[2], J. González[2], L.J. Yebra[1]
[1]CIEMAT, Spain, [2]Aunergy Thermosolar S.L., Spain

Corresponding author: J. Bonilla, CIEMAT-PSA, Research Centre for Energy, Environment and Technology - Solar Platform of Almería, Ctra. Senes s/n, 04200 Tabernas, Almería, Spain, `javier.bonilla@psa.es`

Abstract.

This paper presents a heliostat field simulator, based on a hybrid model, using Modelica as the modelling language. Modelica is both an object-oriented and acausal language. The developed simulator takes advantages of the Modelica features such as: i) object-oriented languages allow developing a set of reusable objects which can be used in future developments; and ii) acausal languages allow describing the behaviour of dynamic systems using models based on differential equation systems. The heliostat field simulator is the union of the hybrid heliostat field model and a wrapped model which handles the real-time simulation and communication issues between the heliostat field simulator and a control system application which is in charge of manipulating and controlling the heliostat field.

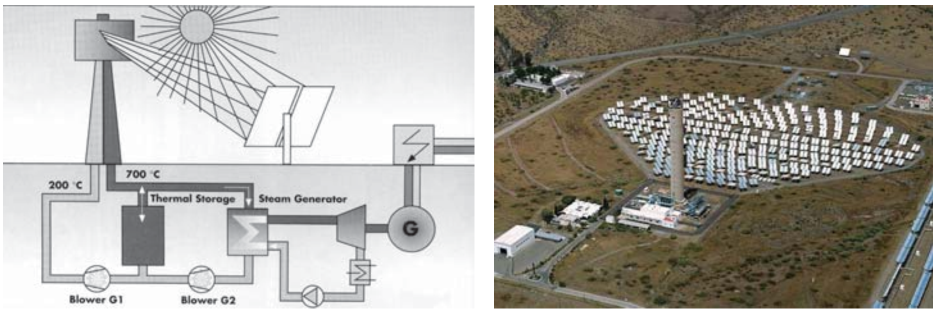
1 Introduction

Design and development of dynamic models for simulation and control system design purposes, is getting importance in solar thermal industrial processes. One example is the deployment of advanced control systems that optimize the overall performance of central receiver solar thermal power plants. It is a fact nowadays that this task is a priority research line [8] at CIEMAT National Labs (Centro de Investigaciones Energéticas, Medioambientales y Tecnológicas - Research Centre for Energy, Environment and Technology), public organism owned by the Spanish Ministry of Science and Innovation.

During the last years a tough effort has been devoted to the development of control systems for solar thermal power plants, making an important part of the experiences directly against the real plant. These real tests have increased the resources needed for the development. Nevertheless, some of these systems, are expensive, scarce and present a costly experimentation time. This fact has motivated the development of a dynamic model for the CESA-I heliostat field plant, aimed mainly as a tool for the enhancement of advanced control algorithms.

2 Central Receiver Solar Thermal Power Plant. CESA-I test facility

In this section an overview of the basic components and operating procedures for a Central Receiver Solar Thermal Power Plant (CRSTPP) and the features of the CESA-I test facility are introduced. Figure 1(a) shows an explicative diagram of a general CRSTPP.



(a) Schematic diagram of a central receiver solar thermal power plant [8]

(b) CESA-I test-bed facility

Figure 1: Central receiver solar thermal power plant

The operation of this kind of plants is based in the concentration of incoming solar energy using a heliostat field that reflects the incident solar radiation onto a (typically volumetric) receiver (theoretically onto an optical point in the 3-D space). As the sun position changes during the day, each heliostat of the field has to change its position in real time according to the selected aiming point on the receiver, as different aiming points can be selected in order to achieve a uniform temperature distribution on the receiver [2]. The receiver is located at the top of the tower (84 m height in CESA-I) and acts as an energy exchanger, receiving solar energy and transferring it to a thermohydraulic circuit with air medium (see Figure 1(a)). The system is also composed of an energy storage tank, an air/water-steam heat exchanger (evaporator), blowers and valves. The combined action of the blowers allows feeding either the storage tank or the heat exchanger with hot air. The evaporator is formed of the primary circuit and a secondary one with subcooled inlet water and with superheated steam outlet. A measurement of the overall concentrated input radiation, a controlled water pump and an outlet controlled valve define the main boundary conditions for the system.

The modelling and simulation activities object of this work are focused on the CESA-I facility, a central receiver solar thermal power plant belonging to CIEMAT, and located at the Plataforma Solar de Almería (PSA), South-East of Spain. This test-bed plant can be seen in Figure 1(b), and it is an experimental prototype for electricity generation among other research projects.

The CESA-I facility collects direct solar radiation by means of a field of 300 heliostats (39.6-m^2-surface) distributed in a 330-x-250-m north field into 16 rows. The heliostats have a nominal reflectivity of 92%, the solar tracking error on each axis is 1,2 mrad and the reflected beam image quality is 3 mrad. The maximum thermal power delivered by the field onto the receiver aperture is 7 MW. At a typical design irradiance of 950 W/m2, a peak flux of 3.3 MW/m2 is obtained. In addition, the 99% of the power is focused on a 4m-diameter circle, 90% in a 2.8-m circle.

3 Object-Oriented and Acausal Modelling with Modelica

Modelica is a standard unified modelling language [4] with many advantages for modelling dynamic systems, because it is both, an object-oriented and acausal language. The object-oriented feature allows developing a set of reusable objects which can be used in future developments and the acausal feature allows describing the behaviour of dynamic systems using the differential equation systems which describe them. Dynamic behaviour and numerical aspects are taken into account in Modelica, because it provides equation sections and event modelling [1].

Moreover, Modelica has a library, *StateGraph*, for modelling hierarchical state machines. The *StateGraph* Modelica library offers features to define conveniently discrete events and reactive systems in Modelica models. Since Modelica is used as an action language, a Modelica translator can guarantee that a StateGraph has deterministic behaviour. StateGraph models can be combined with components of any other Modelica library and can therefore be very easily used to control a continuous plant [5].

To take advantage of all these features, the heliostat field simulator has been developed using Modelica language.

4 Heliostat Field Simulator

The developed heliostat field simulator is mainly the union of the hybrid heliostat field model and a wrapped model which handles the real-time simulation and communication issues. The heliostat field simulator is communicated with a control system application [3] which is in charge of manipulating and controlling the heliostat field. The figure 2 shows the heliostat field simulator and control system application block diagram.

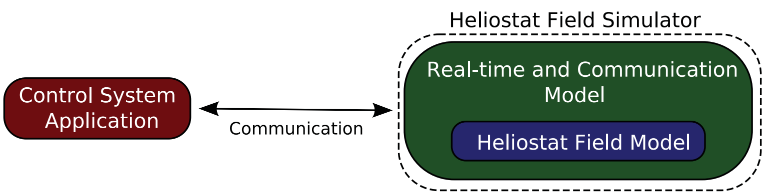


Figure 2: Heliostat field simulator and control system application block diagram

4.1 Heliostat Field Model

In order to obtain the response of a single heliostat, a simple hybrid model has been developed to combine the continuous variables inside the dynamic part of the system with discrete variables that characterize the operation mode. Then, the one-heliostat model may be briefly explained as follows:

- Movement dynamic. The continuous variables of the system are the azimuth and elevation positions which can be obtained through movement equations using the angular velocities as known parameters.
- Operation mode. Not only each single heliostat must achieve the desired reference position defined by the control system application, but also communication failures and timeouts must be taking into account by the heliostat. These characteristics have been included in a state machine using the *StateGraph* library.

Figure 3 shows the one-heliostat component with the following characteristics:

- The input signals are a collection of integer signals that represent the received bytes from the control system application. Notice that these input signals are the same for all the heliostats which belong to the same heliostat row.

- These bytes are decoded to obtain the heliostat identifier (*IH*) parameter (which characterizes univocally each heliostat), the control message (*m*) and the azimuth and elevation references (*ra*, *re*).
- A state machine component, based on the *StateGraph* library, makes possible to know the angular velocities in both directions (*wa*, *we*) which depend on the heliostat state.
- The azimuth and elevation positions (*xa*, *xe*) are calculated with the movement dynamic using the angular velocities provided by the state machine component.

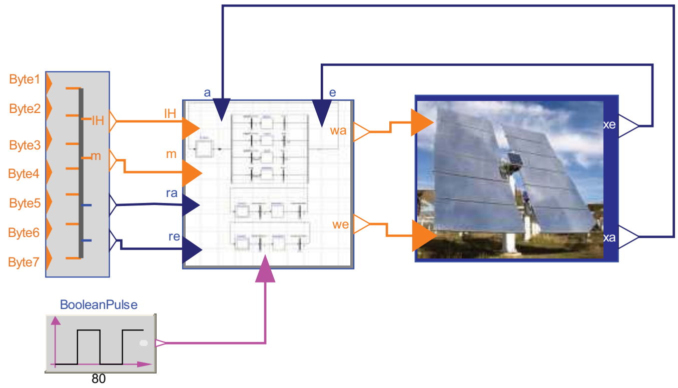


Figure 3: The hybrid one-heliostat model

A heliostat row model is just *NRow* heliostat model instantiations, being *NRow* a parameter that defines the number of heliostats in that row. Notice that the control system application sends input commands to each one of the 16 rows, so that the heliostat field simulator is composed of 16 heliostat-row instantiations.

4.2 Real-time Simulation and Communication Model

The real-time and communication model wraps the main model, the heliostat field model. This wrapped model provides the following functionality:

- Real-time simulation
  - Synchronize the simulation time dynamically to real-time.
- Communication issues
  - Receive input commands from the control system application.
  - Decode input commands.
  - Encode output data from the heliostat field model.
  - Send output data.

Real-time demands that the simulation time must be shorter than the simulated time. Since this condition is a fundamental requirement for models to be used in real-time simulations, in some cases, the complexity of the model must be reduced at the expense of loosing quality.

The communication issues simulate the behaviour of the communication line between the simulator and the control system application involving receive, send, encode and decode messages. The communication line between both applications is simulated by two FIFO files in the operative system for each heliostat row in the field, one for sending and another one for receiving. Message decoding is the process to translate the incoming binary message from the control system application into the corresponding input variables in the Modelica model. On the other hand, message encoding is the reverse process.

4.3 Control System Application

The control system application is in charge of manipulating and controlling each heliostat in the field. The main goals which this application has to achieve are the following:

- It has to be a generic application allowing different kind of heliostats and even allowing different central receiver solar thermal power plants.
- The application has two different subsystems, an intuitive user interface, see Figure 4, and the heliostat field control system.
- It has to be a scalable distributed application. This feature allows that the application can be executed in several computers to achieve scalability and robustness independently of the number of heliostats in the field.

- The application has to allow different communication methods with the heliostat field.
- The application can be connected to both the real plant and the simulated plant.
- It has to be a multi-platform application. The Adaptive Communication Environment (*ACE*) [6] as development framework and *Qt* [7] as user interface framework have been chosen.

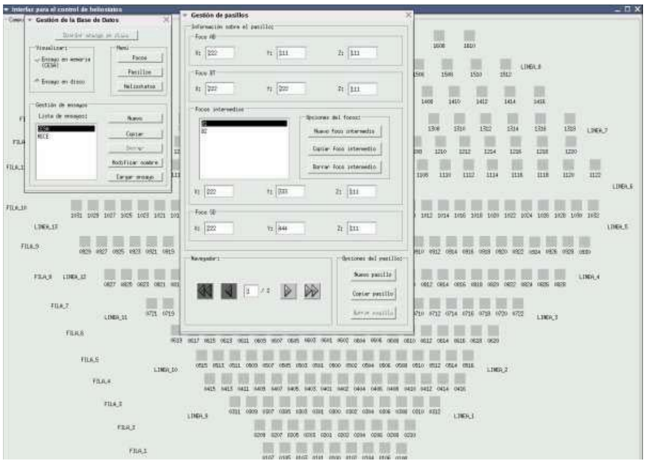


Figure 4: Control system application user interface

5 Conclusions and Future Work

This paper explains briefly a heliostat field real-time simulator developed using Modelica language as the modelling tool. The main goal of the developed model is providing a virtual system, with the same response as the real one, to test the control system application and to reduce the costly real experiments over the real plant.

Future works will include a detailed explanation of real-time, communication and dynamic heliostat models. Moreover, new heliostat models including the incident solar radiation reflections onto the receiver to obtain the temperature distribution, will be implemented. The combination of the movement dynamic, temperature distribution on the receiver, communication model and real-time simulation will provide an efficient and practical tool to test advanced control systems for aim-point tracking to optimize the overall performance of central receiver solar thermal power plants.

6 Acknowledgment

The authors wish to acknowledge the financial support provided for this work by the CIEMAT, and the Spanish Ministry of Science and Innovation, without which this work could not have been done. The authors also gratefully acknowledge the comments of the reviewers.

7 References

[1] Fritzon, P.: *Principles of Object-Oriented Modelling and Simulation with Modelica 2.1*. John Wiley & Sons, IEEE Press, 2004.

[2] García-Martín F.J., Berenguel M., Valverde A., Camacho E.F., *Heuristic Knowledge-based Heliostat Field Control for the Optimization of the Temperature Distribution in a Volumetric Receiver*, In: Solar Energy, vol 66, 1999, 355–369.

[3] Gozález J., Yebra, L.J., Berenguel, M., Valverde, A., Romero, M.: *Real-time Distributed control system for heliostat fields (in Spanish)*. In: XXV Jornadas de Automática, 2004.

[4] Modelica Association.: *Modelica Standard Library*. Online: http://www.modelica.org/libraries/Modelica, 2008.

[5] Otter, M., Årzén, K-E., Dressler, I.: *StateGraph – A Modelica Library of Hierarchichal State Machines*. In: Proc. 4th International Modelica Conference, 2005.

[6] Schmidt D.C.: *The Adaptive Communication Environment (ACE), DOC software*. Washington University, University of California, Irvine, and Vanderbilt University. Online: http://www.cs.wustl.edu/ schmidt/ACE.html, 2007.

[7] Trolltech: *Qt cross-platform application framework*. Online: http://trolltech.com/, 2008.

[8] Yebra, L.J., Berenguel, M. and Dormido, S. and Romero, M.: *Modelling and Simulation of Central Receiver Solar Thermal Power Plants*. In: Proc. 2005 European Control Conference Decision and Control CDC-ECC'05. 44th IEEE Conference, 2005, 7410–7415.