
Robert Pitz-Paal
Deutsches Zentrum für Luft- und Raumfahrt e.V.
Solare Energietechnik, D-51170 Köln, Germany

Scott Jones*
Sandia National Laboratories, Albuquerque, USA

*Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000.
DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.
DISCLAIMER

Portions of this document may be illegible electronic image products. Images are produced from the best available original document.
Abstract. A new approach to modeling solar thermal electric plants using the TRNSYS simulation environment is discussed. The TRNSYS environment offers many advantages over currently used tools, including the option to more easily study the hybrid solar/fossil plant configurations that have been proposed to facilitate market penetration of solar thermal technologies. A component library developed for Rankine cycle, Brayton cycle, and solar system modeling is presented. A comparison between KPRO and TRNSYS results for a simple Rankine cycle show excellent correlation.

1 INTRODUCTION

Performance calculations are necessary to assess the feasibility of proposed solar thermal power plants and to monitor their performance once operational. The SOLERGY code [1], created by Sandia National Labs in the mid-eighties, had become a quasi-standard for estimating the annual performance of solar thermal plants. However, the capabilities and the ease of use of SOLERGY are not up to date with today's standards of computing power and user-friendliness. SOLERGY is limited to energy flow calculations for a hard-wired plant configuration. This has required changes to the FORTRAN source code to adapt to different plant configurations— for instance, LUZERGY was created to handle plants without storage. This complication is now especially restrictive because of the many hybrid cycle concepts combining solar radiation and fossil heat input that have been recently proposed to facilitate the market penetration of solar thermal power plants.

The objective of this paper is to present a more modern and flexible approach using the first version of a solar thermal electric component (STEC) model library for the computer code TRNSYS [2]. This library has been created in joint effort by DLR, IVTAN and Sandia and is now available to all SolarPACES members.

TRNSYS and its user interface IISibat were selected for this objective because they permit the user to use predefined components or create their own FORTRAN component models with defined input and output interfaces that can be easily connected to create arbitrary systems. Models of varying complexity such as simple energy balance models (like SOLERGY), more complex steady-state thermodynamic models, or even simple transient thermodynamic models can be implemented. Moreover, the availability of TRNSYS source code and its reasonable price may increase the acceptance of this software in the solar thermal power community and may help to increase the library’s use by others.

In the following section, the general features of TRNSYS and its user interface IISIBAT are summarized briefly. The major part of this paper is a description of the new STEC model library. Finally, a comparison between the steady state results from the commercially-available KPRO code and from TRNSYS for a simple Rankine cycle is presented.
2. THE TRNSYS COMPUTER CODE AND THE IISIBAT USER INTERFACE

TRNSYS has been under development at the University of Madison, WI, for more than 20 years. Its major field of application is the evaluation of solar systems for heating and cooling purposes and domestic hot water applications. However, the flexibility of the code results in the fact that a worldwide user group has created component models in many other application fields. The current version of TRNSYS, release 14.2, with the IISIBat (graphical) interface was used for this work.

The principle philosophy behind TRNSYS is the implementation of algebraic, and first-order ordinary differential equations describing physical components into software subroutines (called types) with a standardized interface. This interface consists of so called "input" and "output" quantities. Outputs can be either physical quantities or first order derivatives (with time) of physical quantities. The functional relations between inputs and outputs are defined in each subroutine (see Figure 1). To create a system model, the user simply connects the outputs of components with the inputs of other components. The user does not have to worry about how to solve the complex set of equations resulting from the system layout because the TRNSYS kernel performs this function by successive substitution or Powel's Method.

A variety of utility components help to handle input from data files, generate forcing functions, print, plot, integrate or interpolate data. These utilities are linked to the physical components in exactly the same way as described above.

A new user interface named IISibat that was released 3 years ago has significantly improved the capabilities of TRNSYS. Its major features are displayed in Figure 2 and summarized here:

- A graphical user interface using an assembly window in which the components are represented by icons and the connections among them are represented by lines.
- The creation of macro components. Complex systems usually consist of a huge number of components and links, resulting in a confusing graphical representation. To avoid this, there is the option to combine a group of components and their links in a macro, which is displayed as a single icon thereafter. The user can "zoom" in to or out of a macro to display the desired level of detail.
- Assistance with unit conversion and automatic checking that input values are within bounds.
- Advanced administration of users, projects, and components to allow multi-user development in parallel.
- Easy integration of new component models, supported by an icons generator, input masks generator and FORTRAN software templates, as well as documentation support.
- Support for parametric studies with automated batch processing.

These major features of the IISibat graphical interface to TRNSYS help to set-up problems efficiently and with a low probability of making errors.
3. THE STEC MODEL LIBRARY

The new STEC library is based on steady-state energy conservation (1st and 2nd laws) formulated in thermodynamic quantities (temperature, pressure, enthalpy). It consists of models suitable for a Rankine cycle, a Brayton cycle, and a solar system (central receiver, heliostat field, and parabolic trough models). TRNSYS standard components also include models for a storage tank and a rockbed storage. In order to consider transient effects like start-up, an 'inertia' model has been developed that can be linked to each of the previously mentioned components and easily tuned to match empirical data. This permits inertia to be considered only in those components where it has a big impact, and avoids the huge gain in computational complexity of a full transient model. The quasi-steady-state formulation of most component models also simplifies their validation through comparison to steady state results from commercial power cycle codes.

The following sections summarize the steady state models for the solar components and the Rankine cycle. Models for a Brayton cycle have been developed by the Russian colleagues from IVTAN and are described in detail in [5]. The subsequent section deals with the thermal inertia model and how it is mated to the steady state components.

3.1. Modeling Steady State Performance

The most challenging task for the development of new library component models is the definition of the input and output quantities because this has a major impact on flexibility, number of control components required, and computational complexity. Therefore, it is not always the best choice to use physical inlet conditions of a component as TRNSYS inputs and physical outlet conditions as TRNSYS outputs. If we consider, for example, the inlet and outlet quantities for a water-steam kettle evaporator that is heated by a single phase fluid (like flue gas). The choice of water and flue gas inlet temperatures, pressures, and mass flows as inputs and the respective outlet quantities as outputs would result in the problem that both mass flow rates are not independent of each other since the outlet steam is limited to saturated conditions. One would need an additional, external controller that would set a pump speed to meet the outlet conditions. A simpler approach is to define the water flow rate as an output quantity, calling it 'demanded water flow'. This could be connected directly to a pump component, so that the flow rate through all other components connected in series with the evaporator is fixed by this setup and no further controller is needed. This idea of a 'flow demand' as an output is used in several components. A preheater for example demands for a steam flow to the extraction of a turbine, or an attemperator demands for a feed water flow rate.

3.2. Parabolic Trough

The parabolic trough collector based on the model of Lippke[3], uses an integrated efficiency equation to count for the different fluid temperature at the field inlet and outlet of the collector field. It calculates the demanded mass flow rate of the heat transfer fluid to achieve a user-defined outlet temperature $T_{out}$ by

$$
\dot{M} = \frac{\dot{Q}_{net}}{c_p(T_{out} - T_{in})}
$$

using

$$
\dot{Q}_{net} = \dot{Q}_{abs} - \dot{Q}_{pipe}
$$

and

$$
\dot{Q}_{abs} = I \cdot A_{Aperture} \cdot \left[LMS(A + B \frac{1}{2}(\Delta T_i + \Delta T_o)) + C \frac{\sqrt{3}}{I}(\Delta T_i + \Delta T_o) + D \frac{\sqrt{3}}{I}(\Delta T_i - \Delta T_o)^3\right]
$$

The coefficients A, B, C and D are empirical factors describing the performance of the collector. The factor L is the incident angle modifier, M considers end losses and S considers shading of parallel rows. Evaluation of these parameters is described in [3]. $\Delta T_{I,o}$ is the difference between collector inlet or outlet
temperature and ambient temperature and I is the direct normal irradiation. $\dot{Q}_{\text{pipe}}$ accounts for losses in the piping and expansion vessel using empirical coefficients.

The model considers also electrical parasites for tracking, startup and shutdown as well as for pumping. Shutdown is performed automatically at high wind speeds. A minimum flow rate (turn down ratio) can be defined at which the plant is shut down. The fraction of field in track can be specified by the user as a control input.

3.3. Heliostat Field

The heliostat field model requires a user-supplied field efficiency matrix to evaluate the field efficiency, $\eta_{\text{field}}$, as a function of the date and time. The power to the receiver is evaluated by:

$$\dot{Q}_{\text{rec}} = A_{\text{field}} \cdot \rho_{\text{field}} \cdot I \cdot \eta_{\text{field}} \cdot \Gamma$$

using total mirror surface $A_{\text{field}}$, reflectivity $\rho_{\text{field}}$, direct normal radiation $I$ and a control parameter $\Gamma$ describing the fraction of the field in track. The model also considers electrical parasitics for tracking, startup, and shutdown. Shutdown is performed automatically at high wind speeds.

3.4. Power Tower Receiver

The receiver model is like the trough model in that it provides as output the flow rate required to achieve the outlet temperature set point (see eq. (1)):

Since our initial heliostat field model, like SOLERGY, is based upon a simple field efficiency table interpolation, only the total power to the receiver is calculated. To find the detailed flux distribution on the receiver, a complex numerical convolution, or ray-trace optical model, must be used. Without detailed flux information, an empirical receiver heat loss model is more appropriate than one based on heat transfer relations at the receiver's surface. In this model, the conductive losses are neglected in the calculation of the net absorbed power,

$$\dot{Q}_{\text{net}} = \alpha \dot{Q}_{\text{inc}} - \dot{Q}_{\text{conv}} - \dot{Q}_{\text{rad}}$$

where the subscripts $\text{inc}$, $\text{conv}$, and $\text{rad}$ stand for the incident power, the power losses to convection, and the power losses to radiation. The losses are described by third order polynomials with user-supplied coefficients:

$$\dot{Q}_{\text{conv}} = C_0 + C_1 V + C_2 V^2 + C_3 V^3$$

$$\dot{Q}_{\text{rad}} = R_0 + R_1 \dot{Q}_{\text{inc}} + R_2 \dot{Q}_{\text{inc}}^2 + R_3 \dot{Q}_{\text{inc}}^3$$

where $V$ is the wind velocity. The user can also specify the turn down ratio - below which the power is discarded. The receiver efficiency, $\eta = \frac{Q_{\text{net}}}{Q_{\text{inc}}}$ is also provided as an output.

3.5. Boiler

Since boilers may look very different depending on the application, the TRNSYS approach is to use generic heat exchangers elements that can be combined in a macro icon to create a complex system. One-phase and two-phase heat exchangers types are available: The two-phase heat exchanger generates saturated steam and demands for a feed water flow, the one-phase heat exchanger uses inlet conditions of temperature, pressure, and flow rate of the hot and cold side as input quantities. The transferred heat is evaluated using the effectiveness method in both cases [4]. The overall heat transfer coefficient, $U_A$, can be scaled with the cold side mass flow rate using a power law. Also the reference pressure loss may be scaled using a power law with the cold side flow rate.
3.6. Steam Turbine

The steam turbine is assembled using two generic submodels: One for a turbine stage and one for the steam extraction. The turbine stage evaluates the enthalpy at the stage outlet from the inlet conditions using an inner turbine efficiency which can be specified by the user as a function of the flow rate using a third order polynomial. The pressure at the turbine inlet is evaluated from the pressure at the outlet using Stoidola's law of the eclipse. Mechanical losses are considered by a mechanical efficiency. The turbine extraction works as a splitter that receives a flow demand signal (normally from a preheater). The flow is split according to the demand. If the turbine inlet flow rate is below a user-specified limit, a controller redirects it to bypass the turbine assembly completely.

3.7. Preheater/Condenser

Two different preheater types are available: a mixing preheater (also called dearator) and a heat exchanger type. Both demand for a steam flow which is condensed to preheat the water. For the heat exchanger type the effectiveness method is used as discussed in the boiler section. For the mixing type conservation of energy and mass is used.

The condenser model is similar. It assumes a constant temperature difference between condensate and cooling water temperature.

3.8. Modeling Thermal Inertia

Thermal inertia is modeled with a lumped capacity model using an energy balance for a mass with capacity $C$ and initial temperature $T_0$, that is heated by a capacity flow rate $\dot{C}$, with inlet temperature $T_{in}$, outlet temperature $T_{out}$, ambient temperature $T_a$, and steady state temperature $T_{ss}$ using an overall heat transfer coefficient $UA_i$. Convective energy losses to the ambient of this lumped mass are considered by the overall loss coefficient $UA_a$. Energy gains (if any) are considered by the heat rate $Q$. The energy balance can be solved analytically for the capacity temperature, yielding:

$$T(t) = (T_0 - T_{ss}) \cdot \exp\left(-\frac{t}{\tau}\right) + T_{ss}$$  \hspace{1cm} (8)

with

$$\tau = C(UA_i + \dot{C}) \quad \text{and} \quad T_{ss} = \frac{(\dot{C} + UA_i)Q + (UA_a UA_i + UA_a \dot{C})T_{in} + (UA_i \dot{C})T_{in}}{UA_i \dot{C} + UA_a UA_i + UA_a \dot{C}}$$

This results in a fluid outlet temperature:

$$T_{out} = \frac{UA_i T + \dot{C} T_{in}}{UA_i + \dot{C}}$$  \hspace{1cm} (9)

The coefficients used in this model can be either estimated from the component model for which they are used, or if transient measurement data are available, the parameters can be adjusted to meet the time constant $\tau$.

For heat exchanger models, the capacity is linked upstream on the hot side, so that the hot side inlet temperature of the heat exchanger is reduced during startup and less energy is transferred. An evaporator, for instance, will not demand for a feed water flow as long as the hot side temperature is below the boiling temperature of the water.
4. COMPARISON TO THE COMMERCIAL KPRO CODE

In order to verify the steady state performance of the components for the Rankine cycle, the commercial Computer Code KPRO [6] was used to evaluate the thermodynamic design performance data of a fictive Rankine cycle as presented in Figure 3. The results of the TRNSYS code and its deviation from KPRO are presented for temperature, pressure, and mass flow rate at all relevant locations of the plant. The calculation was based on equal overall heat transfer factors for all heat exchanger units, equal reference pressure losses, and the same inner turbine efficiencies, turbine reference mass flow rates and reference pressures. Figure 3 shows that the conformity of both codes is excellent. For the electrical power output, there was a 0.6% difference between the values computed by each code. Slight differences may result from different formulations of the property functions. The function used in TRNSYS is less precise but much faster than the one used in KPRO. An annual performance calculation ran less than 5 minutes on a fast PC (0.03 sec/time step) whereas it took more than 10 sec on KPRO for a single run. KPRO itself is neither able to perform calculations over time (e.g. annual runs), nor consider inertia effects and can not be expanded by user defined components.

5. CONCLUSIONS

TRNSYS offers many advantages over currently-used simulation environments like SOLERGY, including the ability to easily model various hybrid plant configurations. The excellent correlation between KPRO and TRNSYS results for the simple Rankine cycle validates these components of the new model library. Further comparisons to data of solar power plants are envisaged.

REFERENCES