Simulation and Predictive Performance Modeling of Utility-Scale Central Receiver System Power Plants

by

MICHAEL J. WAGNER

A thesis submitted in partial fulfillment of
The requirements for the degree of

MASTER OF SCIENCE
(MECHANICAL ENGINEERING)

At the

UNIVERSITY OF WISCONSIN – MADISON

2008

Approved by

Professor Sanford A. Klein

Professor Douglas T. Reindl

December 12th, 2008

Abstract

Concentrating Solar Power (CSP) and the power tower technology have reemerged as favorable candidates to satisfy the growing demand for both domestic and alternative energy sources in recent years as the price of fossil fuels has increased significantly and climate-change concerns have surfaced. The power tower concept has been successfully demonstrated with steam, air, and molten-salt based receivers in multiple projects since the early 1980's. However, the availability of detailed, fundamentals-based, long-term transient simulation capability is limited. Research presented in this paper is intended to develop a set of tools to address this need.

The modeling capabilities developed for this purpose include the PTGen program and graphical user interface that facilitates the use of the DELSOL3 code written in the 1980's for Sandia National Labs. This tool provides a methodology for plant design and optimization based on a set of user-specified input criteria. Also included are a set of TRNSYS components for the heliostat field, central receiver, Rankine cycle, and plant control incorporating storage for both 2-tank and 1-tank configurations.

The resulting power tower plant model allows the user increased flexibility and a streamlined method for plant design, optimization, and long-term simulation. The model provides modeling capabilities for fossil-fuel hybridization, thermal storage dispatch control, a wide range of Rankine power generation cycle configurations, a wide range of central receiver geometries, and a wide range of heliostat field layouts, among others. A secondary goal of this research is to provide a transient plant model in TRNSYS for the Solar Analysis Model currently in development at the National Renewable Energy Lab in Golden, Colorado.

Acknowledgements

The successful completion of this work was dependent on the contributions of several individuals – none more important than those made by my principal advisors Professor Sanford Klein and Professor Doug Reindl. Sandy's uncommon ability to help me answer my own questions was central to developing my confidence as a researcher and critical thinker, and for that I am grateful. Doug's tremendous practical knowledge and sense for logical incongruence made him an excellent resource over the course of this project. For their guidance, patience, and insight, I am indebted.

I owe thanks to everyone in the Solar Lab, especially to Greg Nellis for inspiring my passion for computational modeling and thermal analysis, and to my office mates Martin, Mike, Matt, Lucas, Scott, and Pete for always being willing to deliberate ideas and concepts (good and bad).

The impetus for this superiorly interesting project was provided by my advisors at NREL Nate Blair and Mark Mehos and to them I offer my gratitude. Their advice was always constructive and the project latitude that they allowed was certainly appreciated. I would also like to thank Greg Kolb at Sandia National Labs, and Diego Arias and Hank Price at Abengoa Solar for their support.

To my wife Katie – without your unending love, your adeptness in preventing me from completely neglecting myself for the sake of scientific advancement, and without your ability to endure countless conversations brimming with excruciating engineering minutiae, I would not have found success.

To my parents and family – your intellectual curiosity and universal interest in engineering have not been lost on me. Finally to my friends, especially Justin, Jared, and Ben – I believe that the best way to always be wrong is to surround yourself with people who think you are right. Thanks to you all, I have not seen this danger.

Table of Contents

List of Tables

List of Figures

Nomenclature

A	Area [m^2]
A_{field}	Total heliostat field mirror surface area [m^2]
A_{helio}	Individual heliostat reflector area [m^2]
Att_{cos}	Cosine attenuation factor [-]
Att_{rec}	Receiver specular reflectance attenuation factor [-]
c	Specific heat capacity [J/kg-K]
c_p	Specific heat capacity (constant pressure) [J/kg-K]
$\dot{C}$	Thermal capacitance rate [W/K]
C_R	Capacitance rate ratio [-]
D_{tube}	The outer diameter of a single receiver tube [m]
ΔAz	Azimuthal distance between heliostats [m]
ΔR	Radial distance between heliostats [m]
η_{field}	Total heliostat field efficiency [-]
$\eta_{thermal}$	Receiver thermal efficiency [-]
F	View factor [-]
g	Gravity constant (9.81 m^2/s)
Gr	Grashof number
h	Convective coefficient, also Enthalpy [J/kg]
H_{helio}	Height of the heliostat reflector [m]
H_{rec}	Receiver height above the heliostat pivot point [m]
I_{bn}	Beam-normal solar radiation [W/m^2]
k	Thermal conductivity [W/m-K]
k_s	Surface roughness [m]
m	Mixed convection coefficient [-]
$\dot{m}$	Mass flow rate [kg/hr]
$\dot{m}_{htf}$	Mass flow rate of the heat transfer fluid [kg/hr]
N_{helio}	Number of heliostats in the field [-]

n_t	Number of receiver tubes per panel [-]
Nu	Nusselt number [-]
P	Pressure [Pa]
P''_{field}	Specific flux density on a receiver panel [W/m^2]
$\dot{Q}$	Energy transfer rate [W]
$\dot{Q}_{helio}$	The total radiation energy reflected by a single heliostat [W]
$\dot{Q}_{inc}$	Total radiation energy incident on the central receiver [W]
ρ_{field}	Field-average mirror reflectivity [-]
$\hat{R}$	The unit vector from a heliostat pivot point to the receiver
R_{cond}	Conductive thermal resistance [K/W]
R_{conv}	Convective thermal resistance [K/W]
Re	Reynolds number [-]
s	Entropy [J/kg-K]
$\hat{S}$	The unit vector from the solar position to a heliostat pivot point
T	Temperature [°C]
T_{htf}	Temperature of the heat transfer fluid [°C]
t_{PSE}	Test statistic
T_s	Receiver Surface Temperature [°C]
UA	Heat exchanger conductance [W/K]
W_{helio}	Width of the heliostat reflector [m]
$\dot{W}_{cycle}$	Work produced by the power cycle [W]
x	The vertical direction on the receiver

Greek Letters

α	Angle between the heliostat surface normal and DNI [deg], also absorptivity of the receiver surface [-]
Δ	Difference between two values

ε	Surface emissivity [-], also Heat exchanger effectiveness [-]
η	Efficiency [-]
θ	Angle [deg], also statistical main effect
μ	Viscosity [Pa-s]
υ	Kinematic viscosity [m^2/s]
π	Pi
ρ	Density [kg/m^3]
σ	Stefan-Boltzmann constant ($5.67x10^{-8}$ W/m^2-K^2)
ϕ	Azimuth angle relative to the central receiver tower [deg]

Subscripts

amb	Ambient
atm	Atmospheric
boil	Boiler heat exchanger
cold	Cold
conv	Convective / Convection
cw	Condenser cooling water
cycle	Referring to the Rankine cycle
demand	Demanded (required)
design	The value under design conditions
dewpt	Dew point
e	Electric (or electricity)
field	Heliostat field
heat	Heat added
helio	Heliostat
hot	Hot
htf	Heat Transfer Fluid
in	Inlet or In
inc	Incident

min	Minimum
mag	Magnitude
max	Maximum
nat	Natural convection
ND	Non-dimensional
out	Outlet or Out
ph	Pre-heater heat exchanger
R	Vector toward the receiver, also Radial position
rad	Radiative / Radiation
rec	Receiver
ref	Reference, also Reflected
S	Vector toward the sun
sh	Super-heater heat exchanger
sky	Sky
steam	Steam in the Rankine cycle
tube	Referring to a single receiver tube
w	Water or Steam

Acronyms

CRS	Central Receiver System
CSP	Concentrating Solar Power
DNI	Direct Normal Irradiation
DOE	Department Of Energy (United States)
EES	Engineering Equation Solver
HTF	Heat Transfer Fluid
INT	Statistical interaction effect
ME	Statistical main effect
NREL	National Renewable Energy Laboratory
OPEC	Oil Producing and Exporting Countries
PS	Plataforma Solar de Almerìa

PSE	Pseudo-standard error (statistical)
PTGen	Power Tower Generation Program
PV	Photovoltaic
SAM	Solar Analysis Model
SEGS	Solar Energy Generating Systems
SERI	Solar Energy Research Institute (now NREL)
STEC	Solar Thermal Electric Components
TRNSYS	Transient Systems Simulation Program

1 Introduction

In the decade following the 1973 oil embargo and subsequent fuel shortages in the United States, elevated petroleum prices accompanied by regional instability within the Oil Producing and Exporting Countries (OPEC) brought domestic energy independence to the forefront of policy discussion. The uncertainty of this period spurred the development of a number of alternative energy technologies, including advancements in renewable energy sources such as photovoltaic and solar thermal technologies. Government financial support was offered to encourage technology development; in 1977 the Solar Energy Research Institute (SERI – later to become the National Renewable Energy Lab, or NREL) was opened in Golden, CO, to further the goal of energy independence through alternative energy sources (NREL, 2008). Several solar technologies reached the testing phase, including the solar power tower and parabolic trough technologies, and the latter would reach commercial deployment in the form of the SEGS plants in southeastern California.

However, as the solar industry was gaining momentum towards becoming a mature and commercially viable energy producer in the mid-1980's, oil and fuel prices dropped rapidly and public interest in alternative energy waned. While the more thoroughly developed parabolic trough concentrating solar technology continued on through the 1980's with the construction of the 2,200 acre SEGS III-IX plants totaling 310 *MWe* (FPL energy, 2008), demonstration projects such as the Solar One 10 *MWe* power tower completed successful test phases without moving into commercial deployment. Figure 1 shows historical oil prices adjusted to 2000 dollars (Energy Information Agency, 2008).

More recently, a combination of urgent issues including concerns about climate change, the United States' destabilizing dependence on oil imports, the prospect of peaking oil production capacity, acceptance of the finite nature of fossil-fuel resources, and a tremendous spike in fuel prices (shown in Figure 2) has renewed interest in alternative, domestic, and renewable energy technologies like concentrating solar power.

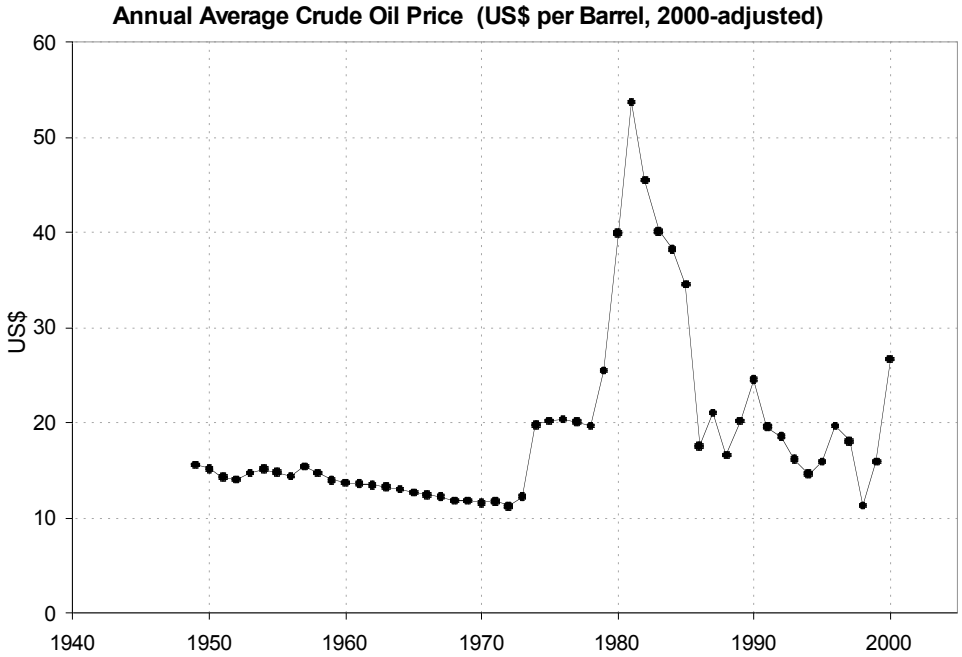


Figure 1: The annual average crude oil price for 1949-2000 in 2000-adjusted dollars (Energy Information Agency, 2008).

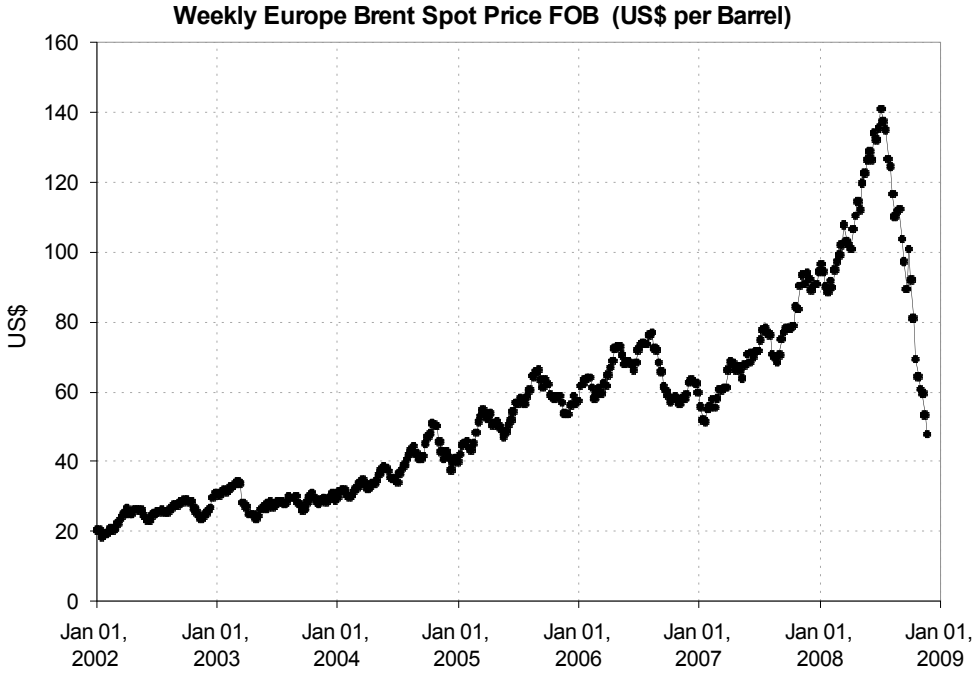


Figure 2: The weekly average Europe Brent Spot crude oil price, indicative of general oil pricing trends (Energy Information Agency, 2008).

The surge in the renewable energy industry has continued despite the recent drop in the price of oil largely due to the increasingly volatile nature of the market and the remaining balance of energy-related issues. However, the long-term viability of alternative technologies like concentrating solar thermal depends on their ability to obtain subsidies, benefit from carbon emission disincentives, or compete economically with traditional power generation industries like coal, natural gas, and nuclear fission. The purpose of the research and analysis presented in the following work is to advance the knowledge and modeling capability for one particular concentrating solar power (CSP) technology – the solar power tower.

1.1 Project Overview

The purpose of the research presented in this thesis is to develop and validate a versatile fundamentals-based model for the power tower CSP technology. This task was undertaken by breaking the plant model concept into a set of sub-models for each individual subsystem in the plant – the tower receiver, heliostat field, power cycle, and thermal storage subsystems. These individual component models work together to predict the behavior of the plant as a whole, and are used for long-term plant simulations in the TRNSYS transient simulation environment.

In addition to modeling the behavior of existing plants, a goal of this research was to provide a tool that is able to determine an optimal plant design given items such as desired electric power production (peak), reflector size, receiver size, and receiver configuration, among others. The ability for the plant designer to analyze a variety of plant configurations and sizes in any location without having previously determined the specific design details is especially desirable since it allows the modeler the opportunity to accurately assess a large number of plant designs in a relatively short amount of time. Thus, both an optimization tool and a long-term performance assessment tool were developed.

A secondary goal of this research is to support the development of the Solar Analysis Model (SAM), which is a new integrated solar technologies analysis software tool currently under development by NREL. This tool will allow NREL, outside researchers, and industry to examine the solar system impacts of various cost, financing and performance decisions and tradeoffs across all solar technologies including concentrating solar power, PV and solar heat. TRNSYS has been selected to provide the some of the system performance calculations within SAM. The TRNSYS engine was chosen because of the extensive existing solar modeling content within TRNSYS for all relevant technologies, the general acceptance of TRNSYS within the solar modeling community, its flexibility and modularity and the fact that it can be re-distributed freely once incorporated into SAM.

1.2 Technology Overview

The central receiver system (CRS) or "power tower" technology is a concentrating solar power application that produces electrical power or facilitates chemical processes my making use of extremely high flux concentrations on the order of one-thousand suns on a relatively small receiver. A heat transfer fluid is used to either absorb heat directly or by contact with the receiver surface; this fluid can be used to power traditional electricity generation cycles like the Rankine or Brayton cycles, or the heat can be used to drive other chemical processes like electrolysis (Kolb, et. al, 2007).

Unlike some prominent alternative energy technologies like wind and photovoltaics, which can only operate when the wind or solar resources are available, solar-thermal technologies like the power tower hold a distinct advantage in that the thermal energy generated by the receiver can be stored for later use. Although solar radiation is still the energy resource, the ability to hold thermal storage to be dispatched during intermittent cloud cover or after sunset reduces the disadvantage of the unpredictability inherent in the use of solar energy and moves the technology closer to the dispatchability of a base-load plant. An illustration of the plant configuration for the Solar II demonstration

project is presented in Figure 3. This plant uses molten salt as a heat transfer fluid, and power is generated with a conventional 11*MWe* Rankine cycle.

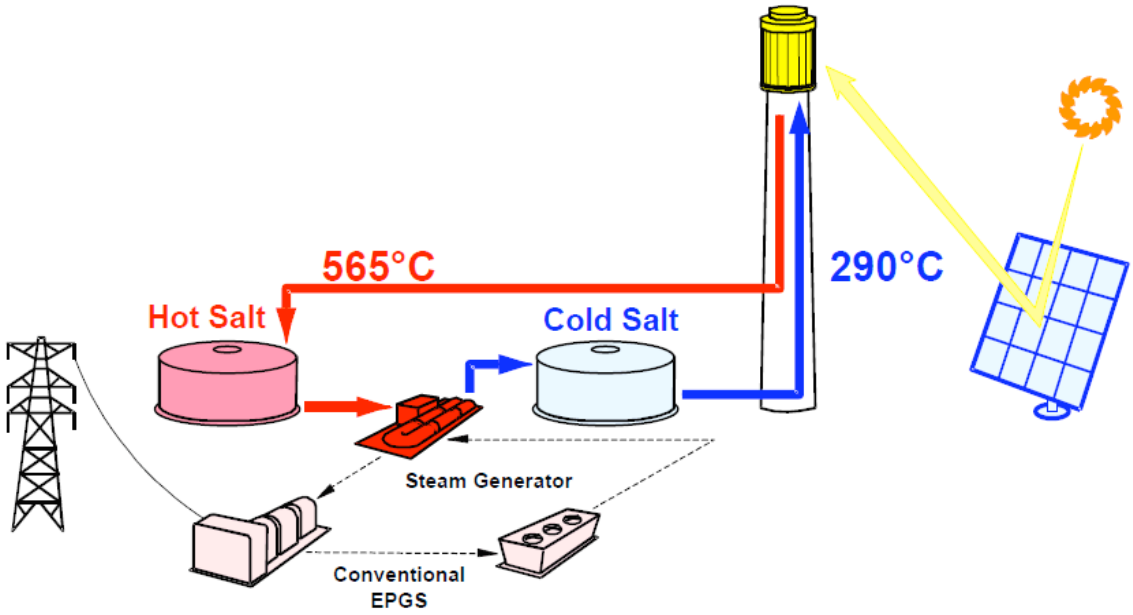


Figure 3: An illustration of the Solar II power tower plant configuration (Pacheco, 2002).

A number of central receiver designs have successfully demonstrated operability. Most of these designs fall within two categories (Karni, et al., 1997): indirect irradiation and direct irradiation. The indirect irradiation configuration (like Solar II depicted in Figure 3) consists of a series of tubular receivers constructed of metal or ceramic tubes through which the working fluid to be heated flows. A number of arrangements are possible, but the outer surface of the tubes generally absorbs the solar radiation and transmits the heat energy to the fluid. For direct irradiation or volumetric receivers, the atmospheric pressure working fluid absorbs the radiation directly or by intimate contact with a solid surface absorbing the radiation directly (Karni, et al., 1997).

The central receivers themselves are not often the major expense when building central receiver-based solar power plant; however, the capital costs of the central receiver can become significant when considering a large-output plant. To leverage the central receiver capital cost, multiple apertures and heliostat fields may be used (Schmitz, et al., 2005).

The heat transfer fluid used as a coolant in the central receivers can vary, but in practice has been steam, air, or molten salt. *Table 1* shows the experimental power towers in existence (not all in operation), and lists each plant's location, output, heat transfer fluid, storage configuration, and operation date.

Table 1: Experimental power towers in the world (Romero, et al. 2002)

Project	Country	Power (MW_e)	Heat Transfer fluid	Storage media	Beginning operation
SSPS	Spain	0.5	Liquid Sodium	Sodium	1981
ERUELIOS	Italy	1	Steam	Nitrate Salt/Water	1981
SUNSHINE	Japan	1	Steam	Nitrate Salt/Water	1981
Solar One	United States	10	Steam	Oil/Rock	1982
CESA-1	Spain	1	Steam	Nitrate Salt	1982
MSEE/Cat B	United States	1	Nitrate Salt	Nitrate Salt	1983
THEMIS	France	2.5	Hitec Salt	Hitec Salt	1984
SPP-5	Russia	5	Steam	Water/Steam	1986
TSA	Spain	1	Air	Ceramic	1993
Solar II	United States	10	Nitrate Salt	Nitrate Salt	1996
Consolar	Israel	0.5 (MW_{th})	Pressurized Air	Fossil Hybrid	2001
PS10	Spain	10	Steam	Steam	2007
PS20	Spain	20	Steam	*	*
Solar Tres	Spain	15	Molten Salt	*	*
**Projects currently in development/construction phase*					

The power cycles commonly used in solar power towers are the Rankine and Brayton cycles. Both can accommodate the high temperatures generated in the solar receiver and are well understood. Clearly, the open Brayton cycle uses air as the working fluid, while closed Brayton cycles can accommodate nitrogen or helium (Forsberg, Peterson, & Zhao, 2007) as the working fluid.

The Rankine cycle is considered in the modeling of the central receiver power plant, and although both cycle types have merit and potential for this application, time constraints prevented development of a modeling tool for direct irradiation receivers. Brayton cycles have generated recent interest in the Power Tower application (Forsberg, et al., 2007) because of the reduced equipment cost associated with the higher pressure cycle. Advances in the high temperature Brayton cycle have spurred nuclear reactor

development programs in the United States, France, Japan, South Korea, and South Africa, and are serving as a model for current research in concentrating solar applications.

Because an important advantage of the Power Tower is the high temperatures capable of being generated within the central receiver, the organic Rankine cycle previously reviewed by McMahan (2006) is an unlikely candidate for power generation. Organic fluids generally diminish in stability at temperatures above 350°C, and high mass flow rates or reduced solar fluxes would likely accompany an organic working fluid.

Storage issues are of practical concern when the thermodynamic fluid may solidify during periods of dormancy. Molten salts generally freeze in the range between 350°C and 500°C, so during extended outages, the system must be drained into a storage tank to avoid solidifying the salt and complicating the plant's restart (Forsberg, et al., 2007). Storage is also useful in providing heat to the power system during adverse weather conditions or at night while electricity demands may require sustained plant opreation. In this situation, the preferred storage medium for salts is graphite because of the relatively low costs ($40/kW) and demonstrated compatibility of fluoride salts and graphite up to temperatures of 1400°C (Forsberg, et al., 2007). In hybrid plants, heat storage is not as important because fossil fuel is used as a backup or supplemental heat source when needed.

1.3 Review of Modeling Tools

The preceding discussion has mentioned two modeling tools in particular – the Solar Analysis Model (SAM) and TRNSYS. This section provides background and a brief summary of the capabilities of the modeling tools used in this research. The major application-specific modeling tools used in this research include TRNSYS, EES, and the Fortran programming language.

1.3.1 The Solar Analysis Model

SAM is not used directly for modeling since one purpose of this research is to support development of a power tower modeling capability to be used by SAM, and so a review of the program is helpful. The SAM program is a joint effort of NREL, Sandia National Laboratory, and the Department of Energy launched in 2006 (NREL, 2008). SAM provides modeling capability for several solar technologies, including the parabolic trough CSP technology, photovoltaic flat plate collectors, and photovoltaic concentrating collectors. Additional technologies are planned for inclusion in the model, including the technology supported by this research, the dish/Stirling parabolic dish CSP technology, and solar heating.

Besides the technology modeling capabilities, a major strength of SAM is the robust financial modeling incorporated into the models. Economic analyses for a range of project magnitudes from residential to utility-scale are supported by the model. Perhaps the most important purpose of this tool is to provide a platform for a consistent methodology in the analysis of multiple solar technologies (NREL, 2008).

1.3.2 TRNSYS

Engineering systems analysis often requires evaluation of system behavior over long periods of time under transient conditions. Renewable energy is transient because of its dependence on resource availability. To accurately predict the performance of systems like the power tower or other CSP technologies that depend on the sun for energy, long-term analysis under varying weather conditions is often required. TRNSYS provides an environment for such analyses, and is organized in such a way that modular component models can be interconnected and evaluated simultaneously. This arrangement is ideal for complex thermal systems like the power tower, since weather data, the Rankine cycle, receiver, heliostat field, and storage models can be produced independently, verified, and connected to form a single large system in TRNSYS.

TRNSYS is an established and respected simulation environment that was originally written in the 1970's at the University of Wisconsin – Madison, and has undergone continual development since. A new program interface was implemented, and commonly used component models are continually added.

The components in TRNSYS are commonly referred to as "Types" and each Type has an associated Type number from 1 to 999. Components are structured to allow for inputs that vary throughout the simulation (called "Inputs"), inputs that are a constant value throughout the simulation (called "Parameters"), and provide outputs. Each component interacts with the other components via the inputs and outputs while providing an interface for code that performs pertinent calculations.

1.3.3 Fortran

Fortran is a general-purpose programming language that is commonly used for engineering and numeric computing tasks. Many versions of the Fortran language have been released since its inception in 1956, but only two of the versions (FORTRAN77 and Fortran90) are used in this research. Fortran was selected as a modeling tool because is particularly well-suited for the computationally intensive and repetitive computing tasks commonly encountered in engineering analysis. Furthermore, many existing TRNSYS components are written in Fortran, and accurate interpretation and improvement of existing work required knowledge of this tool.

1.3.4 EES

The final essential modeling tool employed in this research is the Engineering Equation Solver (EES). This program provides an environment for simultaneous equation solving and assists in engineering analysis by implementing mathematical functions, robust and detailed fluid and material property information, a number of heat transfer correlations for various geometries, plotting capabilities, variable optimization, uncertainty propagation calculation, and many other features. Extensive documentation accompanies the program, making it especially accessible for use in the work presented here. This

invaluable tool was used to develop initial models for various plant components that were later translated into their final version in Fortran. The EES program also originated at the University of Wisconsin – Madison, and its primary author also served as an advisor to this research.

1.4 Review of Existing CRS TRNSYS Components

This section reports on existing components developed in work outside of this research that are related to the CSP power tower technology. The purpose of this discussion is to provide an overview of these components, making note of their positive attributes as well as their shortcomings. Additionally, discussion is provided to analyze the utility of previously developed component models so that unnecessary duplication of effort is avoided. These components were developed by a number of authors for the Solar Thermal Electric Components (STEC) library project, which was initiated in 1998 in a joint effort by DLR (German Aerospace Centre), Sun*Lab/SANDIA and IVTAN (Institute for High Temperatures of the Russian Academy of Science, Russia) (Schwarzbözl, 2006).

1.4.1 Heliostat Field Component

The SolarPACES STEC component library contains heliostat field component (Type 394) constructed by R. Pitz-Paal (DLR, 1997) whose main purpose is to read an input file containing the net field efficiency as a function of solar position, and interpolate the value. Several additional features are included to better model the heliostat field in operation. The number of concentrator units (heliostats) and the power required to track each unit can be entered, allowing the calculation of the total parasitic power required for tracking of the solar position. The ground-level wind velocity is monitored so that in the event that a user-specified maximum wind-speed is reached, the heliostats will defocus and go into a protective “stowed” position. This effectively shuts down the plant in order to protect the heliostat mirror surfaces and the support structure. Additionally, the parasitic power associated with heliostat startup can be modeled.

For receiver models that require the information about the total power from the field to originate from the heliostat component (which is not the case for the receiver model produced by this research since only the total field efficiency value is of use), the mirror surface area parameter can be used. The surface area (A_{field}) is multiplied by the total field efficiency (η_{field}), the incident horizontal beam radiation (I_{bn}), the mirror reflectivity (ρ_{field}), and the fraction of the field that is tracking and not undergoing maintenance, cleaning, or experiencing control problems (Γ) to provide the total power incident on the receiver surface ($\dot{Q}_{inc}$) as shown in Eq.(1.1). If this power is not required, the area can be set as zero.

$$\dot{Q}_{inc} = A_{field} \cdot \rho_{field} \cdot I_{bn} \cdot \eta_{field} \cdot \Gamma \tag{1.1}$$

Certain receiver models (including the receiver model produced in this research) require the distribution of reflected flux on the receiver surface in addition to the total incident power available from the heliostat field. Since the heliostat field component serves only as a mechanism for providing the *total* field efficiency and not the flux distribution on the receiver, central receiver models that require knowledge of an incident flux distribution will conversely have no use for the total reflected power from the field ($\dot{Q}_{inc}$). Receiver components that take the total power from the heliostat field and average the flux over the area of the receiver surface will require the total power from the field (this is the case for the pre-existing receiver components presented in this section).

To specify the total field efficiency, the user provides an input file that indicates the number of azimuth and zenith data points, and this determines the total number of efficiency data points required. For example: four zenith and six azimuth angles would yield 24 data points. The values in this file must be provided in a specific format, and can be determined by a software package such as DELSOL3 or PTGen, or can be entered manually. The PTGen and DELSOL3 codes are discussed in detail in Chapter 5. Table 2 provides a summary of the model parameters, inputs, and outputs for the STEC heliostat field component.

Table 2: The parameters, inputs, and outputs for the STEC heliostat field component (Type 394).

Parameters	Inputs	Outputs
Unit No. of input file	Direct normal insolation	Power to the receiver
Number of zenith angle data points	Wind speed	Defocused power
Number of azimuth angle data points	On/Off control	Parasitic tracking power
Number of concentrator units	Solar zenith angle	Concentrator field efficiency
Mirror surface area	Solar azimuth angle	
Average reflectivity		
Startup energy of unit		
Power to track one unit		
Max allowed wind speed		

Although the existing STEC library heliostat field component provides limited functionality, it was determined that this component should be modified and re-written for this research as an original component. This decision was the result of several factors. The inclusion of the direct normal insolation (DNI) as an input to the existing heliostat field component can be misleading given the setup of the plant model developed in this research, since the efficiency of the field does not depend on DNI levels. Rather, it depends only on the solar azimuth and zenith angles and losses associated with surface quality and tracking. The power output from the field does depend on the DNI; however, the flux distribution unique to each field that is applied to the receiver model contains the appropriate information to calculate the power from the field when paired with the DNI value from a weather file and the net field efficiency from the heliostat field component. This is discussed further in Chapter 2 and Chapter 5.

The component also required modification because an inconsistency in the original code caused the field defocus factor control to be neglected. The heliostat field defocus factor – which may be required when more power is provided from the field than can be used in the power cycle or accommodated by storage – is controlled by the plant control component, discussed below. The final reason that the original STEC component was not used in lieu of a revised version was that the default values provided for the azimuth/zenith number of data points did not match the output of the PTGen program. For the sake of simplicity and intuitive use of the component for this research, modification was desired.

These issues were resolved in the new component written for this research, which has been given type-number Type 221. The new heliostat field component including these modifications is discussed in detail in Chapter 2.

1.4.2 Central Receiver Components

The STEC library contains two central receiver models that can accommodate molten salt or another liquid as a heat transfer fluid, and one air-receiver model. The salt/liquid models will be reviewed in this discussion.

Tower Receiver (Type 395)

The first tower receiver component, which was authored by S.A. Jones of Sandia National Labs (1997), is a simplified model of the central receiver, relying on a receiver thermal efficiency as an input value instead of calculating it directly. This model provides the flow rate required to achieve a temperature set-point as an output. Conduction from the tower to the ground is neglected, and losses due to convection and radiation are described by third order polynomials with user-supplied coefficients. The requirement that these parameters be provided to the model is a major limitation, and contains use of this component to quick analysis of operating power tower plants. The parameters, inputs, and outputs are provided in Table 3.

Table 3: The parameters, inputs, and outputs for the SolarPACES STEC tower receiver component (Type 395).

Parameters	Inputs	Outputs
Tower efficiency	Incident power	Flow rate demand
	Fluid inlet temperature	Fluid outlet flow rate
	Fluid inlet flow rate	Fluid outlet temperature
	Fluid outlet pressure	Fluid inlet pressure
	Temperature set point	
	Fluid specific heat	

The model formulation is relatively simple and is contained within a few lines. The required heat transfer fluid mass flow rate is calculated with an energy balance as shown in Eq.(1.2).

$$\dot{m}_{htf,demand} = \frac{\dot{Q}_{inc} \cdot \eta_{rec}}{\left(c_{htf} \cdot \left(T_{htf,hot} - T_{htf,cold}\right)\right)} \tag{1.2}$$

The variables are defined such that:

$\dot{Q}_{inc}$ = incident power on the receiver

η_{rec} = tower thermal efficiency

c_{htf} = heat transfer fluid specific heat

$T_{htf,hot}$ = HTF outlet temperature set point

$T_{htf,cold}$ = HTF inlet temperature

$\dot{m}_{htf,demand}$ = Resulting mass flow rate demand

Alternatively, the outlet fluid temperature is calculated when the heat transfer fluid mass flow rate is specified as:

$$T_{htf,hot} = T_{htf,cold} + \frac{\left(\dot{Q}_{inc}.\eta_{tower}\right)}{\dot{m}_{htf,demand}.c_{htf}} \tag{1.3}$$

Although this component may be useful in cases where a rough estimate of plant performance with constant receiver efficiency is adequate, this model is not appropriate for the detailed predictive analysis required for this research since it requires as an input the tower thermal efficiency.

Central Receiver, Variable Efficiency (Type 495)

Type 495 was authored by J.M. Crespo of Ciemat (2000) as an enhancement to the previously described central receiver component (Type 395). This model takes variable tower thermal efficiency into account by scaling it with the ratio of the incident power (C_{oc}) to the receiver design thermal power. This ratio and the ratio of the part-load

efficiency to the design efficiency (C_{eff}, set by the user) are correlated using a hyperbolic fit. Two additional parameters (*a* and *b*) are calculated to aid in the process.

$$C_{oc} = \frac{\dot{Q}_{partload}}{\dot{Q}_{design}} \tag{1.4}$$

$$C_{eff} = \frac{\eta_{partload}}{\eta_{design}} \tag{1.5}$$

$$a = \frac{\left(1 - C_{oc}\right) \cdot \eta_{partload}}{C_{eff} - C_{oc}} \tag{1.6}$$

$$b = \left(\frac{a}{\eta_{design}}\right) - 1 \tag{1.7}$$

After calculating the coefficients, the ratio (W_{pl}) of the incident power (converted to the correct units by dividing by $3600 \frac{W - hr}{kJ}$) to the design incident thermal power is calculated and the overall efficiency is determined.

$$W_{pl} = \left(\frac{\dot{Q}_{inc}}{3600}\right) \frac{1}{\dot{Q}_{design}} \tag{1.8}$$

$$\eta_{tot} = \frac{a \cdot W_{pl}}{b + W_{pl}} \tag{1.9}$$

Although this arrangement takes a different approach than the original central receiver model, it likewise relies on empirical relationships to determine the tower behavior instead of providing fundamental analysis. Additionally, the coefficients needed in these empirical relationships are not available without previous knowledge of a specified system. For these reasons, this component was not regarded as a satisfactory model for the purposes of this work. However, this component may also be useful as a rough

analysis tool for well-defined systems, but is not useful in predictive modeling. The parameters, inputs, and outputs for the TRNSYS component are summarized in Table 4.

Table 4: The parameters, inputs, and outputs for the STEC central receiver model (Type 495).

Parameters	Inputs	Outputs
Mode	Incident power	Outlet mass flow rate
Design power	Inlet temperature	Excess mass flow rate
Receiver design efficiency	Inlet flow	Outlet temperature
Part-load power	Inlet pressure	Outlet pressure
Receiver part-load efficiency	Set point temperature	Energy transfer
	Maximum receiver temperature	Excess energy
	Fluid specific heat	Efficiency
	Receiver pressure drop	

1.4.3 Rankine Cycle Components

The STEC library also contains a collection of components that can be used to assemble a power cycle to match a wide variety of Rankine cycle or Brayton cycle configurations. This discussion covers the Rankine steam cycle components. Models are available for five major equipment categories:

- the condenser and condenser pump for condensing functions
- the deaerator, pre-heater, and sub-cooler for pre-heating functions
- the economizer/super-heater and evaporator for steam generating functions
- a variety of splitters and mixers, a turbine stage, a turbine control, and a throttling valve for turbine functions
- other utility components for various purposes (e.g., lumped mass to simulate thermal capacity of components, an electric generator, etc.)

These components, which were authored by R. Pitz-Paal and P. Schwarzbözl of the DLR (1998), can be substituted for the Rankine cycle component developed for this research if desired. However, these components were not used for this research for several reasons. First, several components lacked the detail and predictive capacity desired for the purposes at hand. This is discussed in more detail on a component-by-component basis below. Second, the limited ability of TRNSYS to calculate fluid properties (especially during phase change) was troubling, since the phase change of steam to water is a driving

principle for the Rankine cycle. Finally, a comprehensive and autonomous Rankine cycle model was sought to simplify and standardize the power cycle model for the power tower system.

A brief review of selected components is presented below to highlight the premises on which these components are constructed. Each review contains the parameters, inputs, and outputs for the TRNSYS component, the major equations used to develop the model, and a short summary of the component. For more detailed information on these components and their best use, see Schwarzbözl (2006).

Type 315: Economizer, Super-heater for Water/ Steam heated by One Phase fluid

Table 5: Parameters, inputs, and outputs for STEC library Type 315

Parameters	Inputs	Outputs
Counter-flow mode	Hot side inlet temperature	Hot-side outlet temperature
Overall heat transfer coefficient of exchanger	Hot side flow rate	Hot-side flow rate
Reference press loss cold side	Cold side inlet temperature	Cold-side outlet temperature
Reference cold side flow	Cold side flow rate	Cold-side flow rate
Power law exp for UA	Cold side quality	Heat transfer rate
Power law exp for DP	Cold side outlet pressure	Effectiveness
	Hot side specific heat	Cold side Outlet quality
		Cold Side Inlet pressure

This component is based on equations commonly used to describe the behavior of counter-flow heat exchangers. First, the derived effectiveness relationship is shown (Schwarzbözl, 2006).

$$\eta_{ECO} = \frac{1 - \exp\left(-\frac{UA}{\dot{C}_{\min}} \cdot \left(1 - \frac{\dot{C}_{\min}}{\dot{C}_{\max}}\right)\right)}{1 - \frac{\dot{C}_{\min}}{\dot{C}_{\max}} \cdot \exp\left(-\frac{UA}{\dot{C}_{\min}} \cdot \left(1 - \frac{\dot{C}_{\min}}{\dot{C}_{\max}}\right)\right)} \tag{1.10}$$

In this equation, $\dot{C}_{\min}$ and $\dot{C}_{\max}$ represent the minimum and maximum thermal capacitance rates of the two fluids passing through the heat exchanger. The relationship for UA depends on the ratio of the cold-side fluid mass flow rate to its reference mass flow rate,

scaled by an exponent. This exponent was set as 0.8 in this research (see section 4.2), and a similar value may be used for this model.

$$UA = UA_{ref} \cdot \left(\frac{\dot{m}_{cold}}{\dot{m}_{cold,ref}} \right)^{UA\_\exp} \tag{1.11}$$

UA_{ref}, $\dot{m}_{cold,ref}$, and UA_exp can be specified by the user, and pressure loss is scaled in a similar way where the user has the freedom to specify a satisfactory exponent.

$$\Delta p = \Delta p_{ref} \cdot \left(\frac{\dot{m}_{cold}}{\dot{m}_{cold,ref}} \right)^{\Delta p\_\exp} \tag{1.12}$$

Type 316: Evaporator

Table 6: The parameters, inputs, and outputs for the STEC Type 316 Evaporator.

Parameters	Inputs	Outputs
overall heat transfer factor	Hot side inlet temperature	Hot side outlet temperature
blow down fraction	Hot side flow rate	Hot side outlet flow rate
reference pressure loss	Cold side inlet temperature	Cold side outlet temperature
reference flow rate	Cold side outlet pressure	Cold side inlet pressure
power lax exp for UA	Cold side inlet quality	Cold side outlet quality
power law exp for dp	Hot side specific heat capacity	Cold side flow rate demand
		Cold side outlet flow rate
		Transferred Power
		Effectiveness

In an arrangement similar to Type 315, the evaporator is approached analytically and uses the fluid conductances to determine the effectiveness. The fluid on the cold side of the heat exchanger is assumed to be steam, and the specific heat of the fluid on the hot side is based on the working fluid used for the cycle. An inlet water flow rate is modulated to match the steam evaporation rate within the unit. Conductance is calculated as shown above in Eq.(1.11). The evaporator effectiveness is calculated in Eq.(1.13) and the corresponding heat transfer across the heat exchanger is determined in Eq.(1.14).

$$\eta_{Evaporator} = 1 - \exp\left(\frac{-UA}{\dot{m}_{hot} \cdot c_{p,hot}} \right) \tag{1.13}$$

$$\dot{Q}_{trans} = \eta_{Evaporator} \cdot c_{p,hot} \cdot \dot{m}_{hot} \cdot \left(T_{hot,in} - T_{saturated} \right) \tag{1.14}$$

Turbine Stage (Type 318)

Table 7: The parameters, inputs, and outputs for the STEC Turbine stage Type 318

Parameters	Inputs	Outputs
Design inlet pressure	Turbine outlet pressure	Turbine inlet pressure
Design outlet pressure	Turbine inlet flow rate	Turbine outlet flowrate
Design flow rate	Turbine inlet enthalpy	Turbine outlet enthalpy
Design inner efficiency	Bypass indicator	Turbine power
Generator efficiency		Outlet bypass indicator
Coef. for inner eff eq		Turbine inner efficiency
b coeff for inner eff		
c coeff for inner eff		

This component models the pressure drop across the turbine using the relationship proposed by Stodola and Lowenstein (1945). The inlet pressure is calculated by considering the relationship between the outlet pressure, the steam mass flow rate, the reference (design) values of inlet and outlet pressure and mass flow rate, as shown in Eq.(1.15).

$$P_{in} = \sqrt{\left(\frac{\dot{m}_{in}}{\dot{m}_{ref}} \right)^2 \left(P_{in,ref}{}^2 - P_{out,ref}{}^2 \right) + P_{out}{}^2} \tag{1.15}$$

The behavior of this single-stage turbine is characterized using an isentropic efficiency, which is calculated from a reference value and adjusted based on the mass flow rate of steam through the turbine relative to a reference mass flow rate with the relationship in Eq.(1.16). The coefficients α, β, γ are set by the user, but since these are often not known a priori without performing simulations or equipment testing, this is a major limitation to the turbine model.

$$\eta_{in,turbine} = \eta_{in,ref} \cdot \left(1 + \alpha \cdot \dot{m}_{ratio} + \beta \cdot \dot{m}_{ratio}{}^2 + \gamma \cdot \dot{m}_{ratio}{}^3 \right) \tag{1.16}$$

The ratio of the mass flow rates is given by:

$$\dot{m}_{ratio} = \frac{\dot{m}_{in}}{\dot{m}_{ref}} - 1 \tag{1.17}$$

The turbine can also be combined with a splitter component (Type189) and additional instances of the turbine stage to assemble an extraction turbine.

Condenser (Type 383)

Type 383 models a water cooled condenser with a fixed cooling water temperature rise that is supplied by the user as a parameter. The temperature difference between cooling water outlet temperature and condensing temperature is given by an additional parameter. Therefore, this component assumes that the condensing pressure only depends on the feed water inlet temperature and is constant when this inlet temperature is constant. This limitation neglects other important parameters like the condensing pressure and requires previous knowledge of the condensing temperature, making it nearly useless as a predictive model.

Table 8: Parameters, inputs, and outputs for the STEC Condenser Type 383.

Parameters	Inputs	Outputs
dT Cool water out + condensing temp	Cooling water inlet temp	Condensing Temperature
Temp increase in cool. water	Steam enthalpy inlet	Condensing pressure
	Steam mass flow rate	Transferred power
	Condensate inlet flow rate	Cooling water outlet temp
	Condensate inlet temperature	Cooling water flow rate
	Condensate inlet quality	Condensate flow rate
	Cooling water inlet temp	Condensing Temperature

The heat transfer across the condenser is calculated by:

$$\dot{Q}_{cond} = h_s \cdot \dot{m}_s + h_c \cdot \dot{m}_c - \left(\dot{m}_s + \dot{m}_c\right) \cdot h_{sat} \tag{1.18}$$

where:

h_s = main steam inlet enthalpy

$\dot{m}_s$ = main steam flow rate

h_c = additional condensate inlet enthalpy

$\dot{m}_c$ = additional condensate inlet flow rate

h_{sat} = enthalpy of saturated water at P_{cond}

With the heat flow rate determined, the cooling water flow rate is evaluated.

$$\dot{m}_{cool} = \frac{\dot{Q}_{cond}}{C_{p,w} \cdot \Delta T_{c,w}} \tag{1.19}$$

where:

$\dot{m}_{cool}$ = flow rate of cooling water in

$C_{p,w}$ = specific heat of water

$\Delta T_{c,w}$ = user specified parameter: the temperature increase of the cooling water across the condenser

The condenser pressure is calculated based on the saturation conditions at the condensing temperature. This condensing temperature is determined by evaluating the sum of the temperature increase of the cooling water and the temperature difference between cooling water outlet temperature and condensing temperature. Since all of these values are entered by the user, the temperature is essentially (and thus the pressure) of the hot-side fluid in the condenser are fixed.

This component evaluates the required cooling water mass flow from an energy balance. It may be argued that the temperature increase of the condenser water should be provided to the model as a TRNSYS Input rather than a Parameter so that it can be changed during the course of the simulation based on the simulation of the cooling tower or other cooling apparatus.

It should also be noted that this model does not consider the characteristics of the condenser such as UA and heat transfer rates. This is a major limitation of this model, as

it does not actually *model* the process, but serves as a place-holder for a condenser whose operating outputs are known. As previously indicated, this arrangement could be realistic if a cooling tower with appropriate flexibility were coupled with this system.

2 The Heliostat Field

A noted advantage of the concentrating solar power (CSP) tower technology compared to other CSP technologies such as parabolic trough or linear Fresnel is that the receiver operates at a higher thermal efficiency due to the high incident flux concentration on the receiver surface. But unlike other CSP technologies, the power tower concept requires reflected solar radiation from the sun-tracking mirrors (heliostats) to travel significant distances to a tower-mounted receiver. This distance can be on the magnitude of 1 *km* or more for large plants (Sánchez and Romero, 2005).

Since the average distance between a heliostat and the receiver is considerable, precise construction, installation, and control of the heliostats is required to ensure that optical losses are minimized. This requirement results in the heliostat field capital cost being disproportionately large with respect to the overall plant cost. Ortega, et al., (2006) note the heliostat field capital costs ranges from 30-40% of the total plant capital costs. As a result, careful optimization of the capital-intensive heliostat field is essential for an economically viable power tower system.

A thorough understanding of the mechanisms affecting heliostat field performance is beneficial in achieving an optimized heliostat field layout (where the power reflected to the receiver is optimized in terms of capital cost per unit area of mirror surface). This chapter discusses the theoretical background behind these mechanisms and reviews techniques applicable to optimize the field layout. The resulting tool for applying heliostat field performance to the larger plant model in TRNSYS is also presented.

2.1 Heliostat Field Performance Background

The central receiver system heliostat field contains a varying but generally large number of individual heliostats. The number of heliostats is dependent on the size of individual heliostats and the desired system thermal power. The Solucar/Abengoa PS10 11MWe plant in Sanlúcar la Mayor, Spain, operates with a north-based field containing 624 heliostats at 120 m^2 each (SolarPACES, 2007), while the Solar II 10MWe plant in Daggett, CA, contained a field of 1926 heliostats, 1818 of which were 39.1 m^2 each and 108 of which were 95 m^2 each for a total reflector area of 81,343 m^2 (Pacheco, 2002). Heliostat geometry can also vary widely, with height, width, number of cant panels (the individual facets of each heliostat) – both circular and rectangular variations are possible. Two heliostat designs are shown in Figure 4.

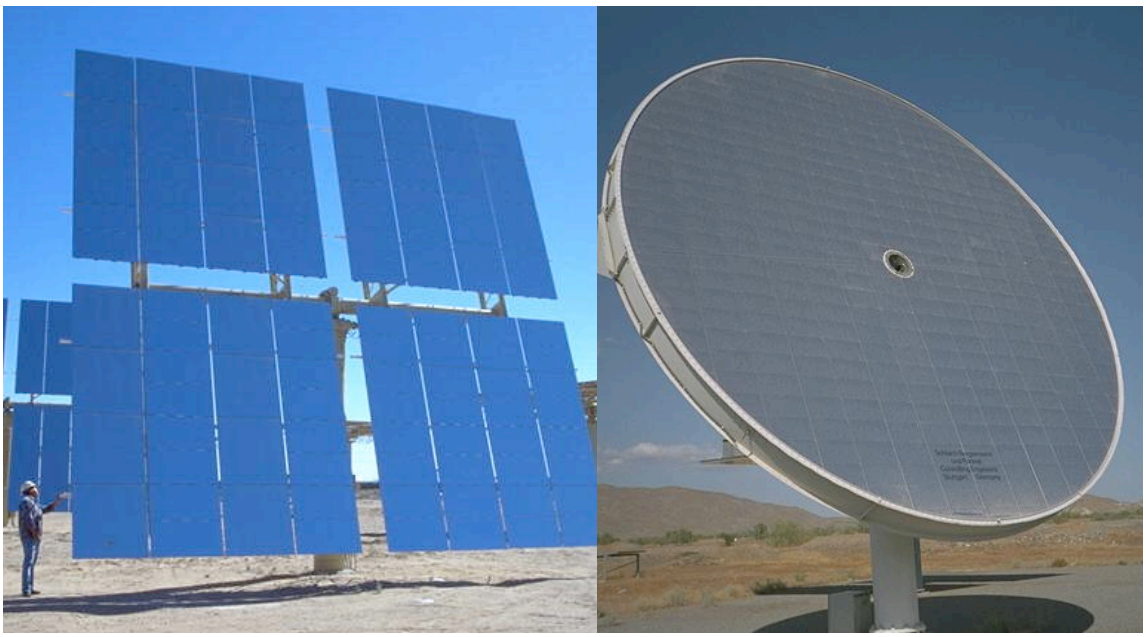


Figure 4: Two heliostat designs. The Lugo heliostat (LEFT) tested at Solar II, and the SBP stretched-membrane heliostat (RIGHT) tested in Almeria, Spain (Sandia, 1997).

Regardless of heliostat geometry, the heliostat is subject to a number of optical losses that result in a reflected image on the receiver surface that is somewhat less than the amount of solar energy originally incident on the surface of the heliostat. The following subsections discuss loss mechanisms and demonstrate layout techniques designed to

overcome the losses to the greatest possible extent. Certain layout patterns emerge as a result, and these common field configurations are also presented.

2.1.1 Losses from the Heliostat Field

A number of losses can adversely affect the performance of the heliostat field including:

- cosine effects
- mirror-to-receiver incidence angle attenuation
- atmospheric scatter
- shading
- blocking
- spillage
- maintenance (some heliostats are necessarily unavailable due to regular cleaning and maintenance requirements)
- field tracking and controller accuracy, including specifically (Stone and Jones, 1999):
    - azimuth rotational axis tilt
    - control system granularity (resolution)
    - atmospheric refraction
    - bending of the heliostat structure due to gravity or wind
    - mirror alignment or "canting"
    - sun position algorithm error
    - computation time error
    - transmission time error

Since some of these items are dependent on the position of the heliostat in the field relative to the tower, it is clear that not all heliostat locations provide equal energy to the receiver over the course of operation. In order to evaluate the potential energy production capability for each heliostat in the field, these losses must be individually modeled as a function of solar position.

Cosine Attenuation

The first and most significant loss to be discussed is attenuation due to the angle between the incident solar beam radiation and a vector normal to the surface of the heliostat. This is called the *cosine effect*, since the radiation reflected to the receiver by the heliostat mirror is proportional to the cosine of the angle in question. To evaluate the surface normal vector, two vectors are required: the first is a vector that intersects the heliostat at its mounting position and is collinear with the solar beam radiation, and the second is a vector traversing the path between the heliostat mounting position and the desired image location on the receiver surface. The face of the heliostat must then be aligned so that the vector normal to the heliostat surface bisects the angle formed by these two vectors.

As the angle between the surface normal and the incident radiation (α) grows, the size of the reflected image decreases and vice versa as the magnitude of the angle decreases. This is shown in Eq.(2.1) where the corresponding reflected radiation ($\dot{Q}_{helio}$) is equal to the cosine of the angle difference (α) times the incident beam radiation (I_{bn}) and the heliostat surface area (A_{helio}).

$$\dot{Q}_{helio} = \cos(\alpha) \cdot I_{bn} \cdot A_{helio} \tag{2.1}$$

A heliostat field that takes this effect into account will place the heliostats such that they are positioned in a way that minimizes the incident angle difference α over the course of the year. For power tower plants in the northern hemisphere where the sun is predominantly (if not exclusively) in the southern sky throughout the year, heliostats positions north of the receiver will reflect more radiation than those positioned south of the receiver. Conversely, power tower plants in the southern hemisphere will contain heliostats that are more effective when located south of the receiver, and plants near the equator will have equally balanced heliostats to the north and south. A figure illustrating this effect is presented by Stine and Geyer (2001), and is reproduced below in Figure 5.

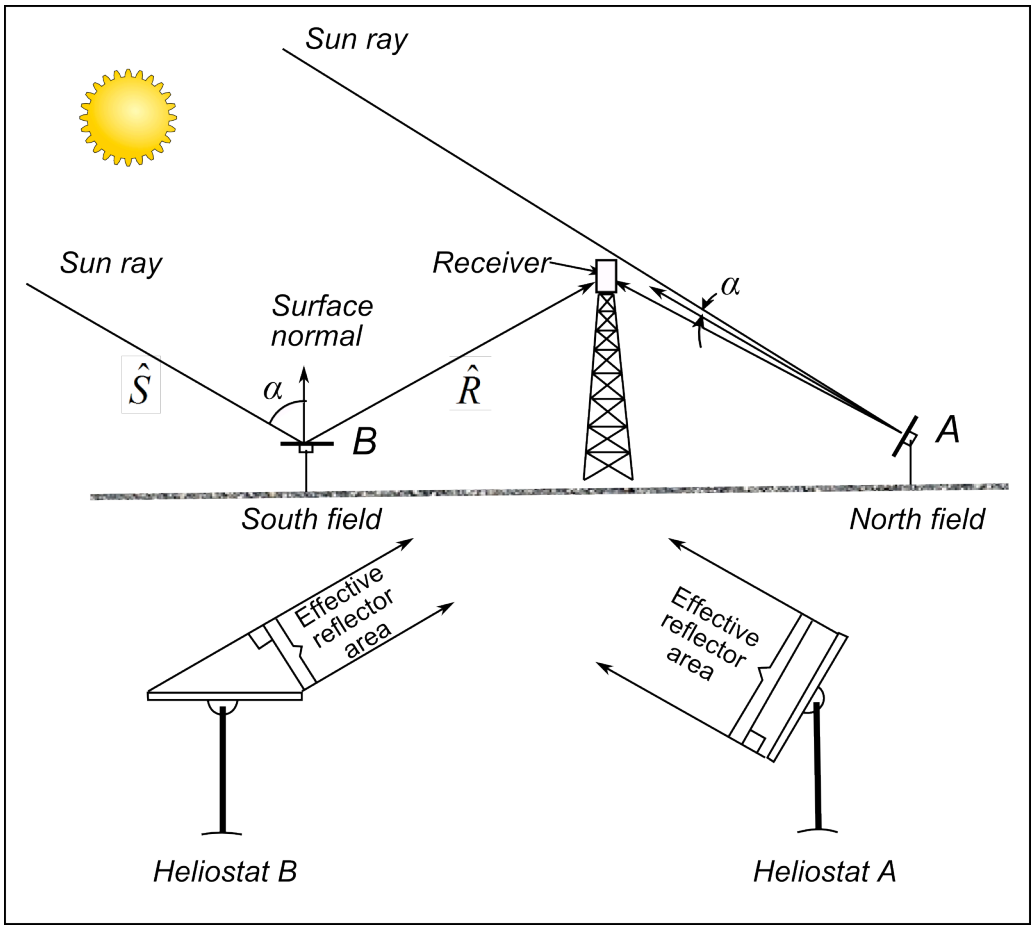


Figure 5: The cosine effect for CRS heliostats (adapted from Stine and Geyer, 2001)

The cosine attenuation is given by Equation(2.2), where the argument α can be evaluated as in Equation(2.3).

$$Att_{\cos} = \sin(\alpha) \tag{2.2}$$

$$\alpha = \frac{1}{2}\left[180 - \frac{\hat{i}_S \cdot \hat{i}_R + \hat{j}_S \cdot \hat{j}_R + \hat{k}_S \cdot \hat{k}_R}{R_{mag} \cdot S_{mag}}\right] \tag{2.3}$$

The $\hat{i}$, $\hat{j}$ and $\hat{k}$ components represent the scalar vector components in the *x*, *y* and *z* directions, respectively. The subscript *S* refers to the solar position vector relative to the heliostat in question, while the subscript *R* denotes the position vector of the heliostat relative to the receiver as shown in Figure 5. This is effectively the same vector path as

the reflected irradiation. Parameters S_{mag} and R_{mag} are defined as the magnitude of the unit vectors $\hat{S}$ and $\hat{R}$ such that the relationship in Eq.(2.4) is valid.

$$\hat{S} = \frac{i_S + j_s + k_s}{S_{mag}}$$
$$\hat{R} = \frac{i_R + j_R + k_R}{R_{mag}} \tag{2.4}$$

Reflective losses from the receiver

The second loss associated with the field is the reflection of radiation from the receiver surface. The proportion of the radiation incident on the receiver surface that is reflected depends on the absorptivity of the receiver surface coating and on the incidence angle of the radiation striking the surface. As the incident radiation deviates from the surface normal, the amount of radiation absorbed decreases to zero at an incident angle parallel to the surface. This relationship is described with Eq.(2.5) (Duffie and Beckman, 2006), and is illustrated in Figure 6 for angles ranging from normal (90°) to parallel with the receiver surface (0°).

$$att_{rec,i} = \begin{pmatrix} 1 - 1.5870\times10^{-3}\cdot\gamma_i + 2.7314\times10^{-4}\cdot\gamma_i^{\,2} - 2.3026\times10^{-5}\cdot\gamma_i^{\,3} + \\ 9.0244\times10^{-7}\cdot\gamma_i^{\,4} - 1.8\times10^{-8}\cdot\gamma_i^{\,5} + 1.7734\times10^{-10}\cdot\gamma_i^{\,6} - \\ 6.9937\times10^{-13}\cdot\gamma_i^{\,7} \end{pmatrix} \tag{2.5}$$

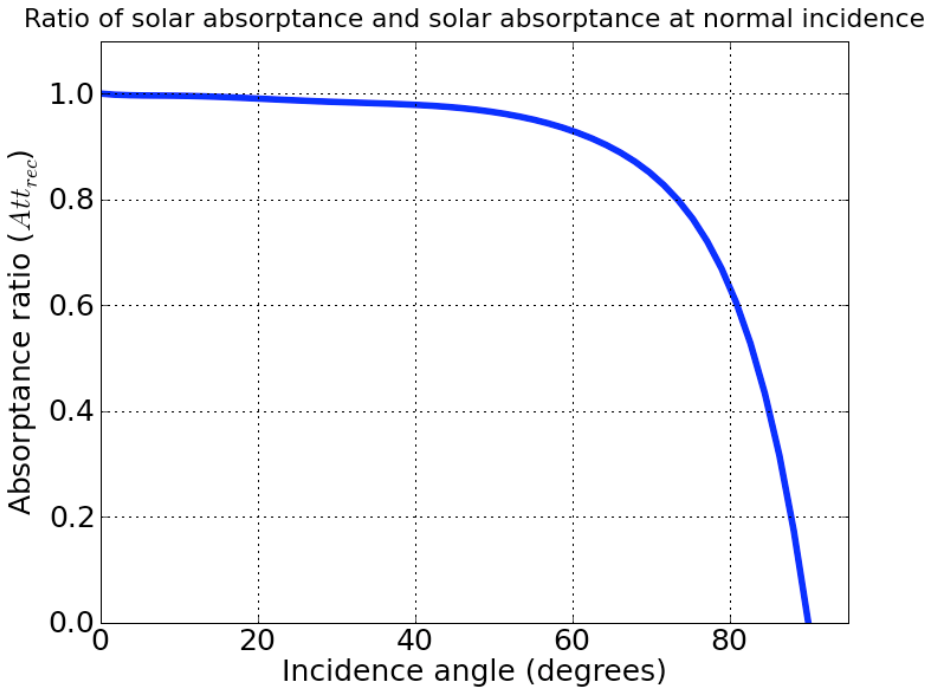


Figure 6: The dependence of absorptivity of a black surface on radiation incidence angle.

The attenuation factor in Figure 3 is a ratio the absorptance at normal incidence and thus is independent of the receiver surface. The reflection losses are compounded by the receiver surface hemispherical reflectivity. A known reflectivity loss as a function of receiver incidence angle allows a portion of the reflection losses to be calculated for each heliostat in the field based on geometry, and regardless of the receiver surface treatment. A contour plot of the field efficiency with only the cosine and incidence angle effects taken into account appears in Figure 7. The plot is generated for a solar zenith of 18.5° and a solar azimuth of 250° so that the solar position is in the southwestern sky relative to the plant. The plot is scaled in terms of tower height, and the field is shown for a range of four tower heights extending in each cardinal direction.

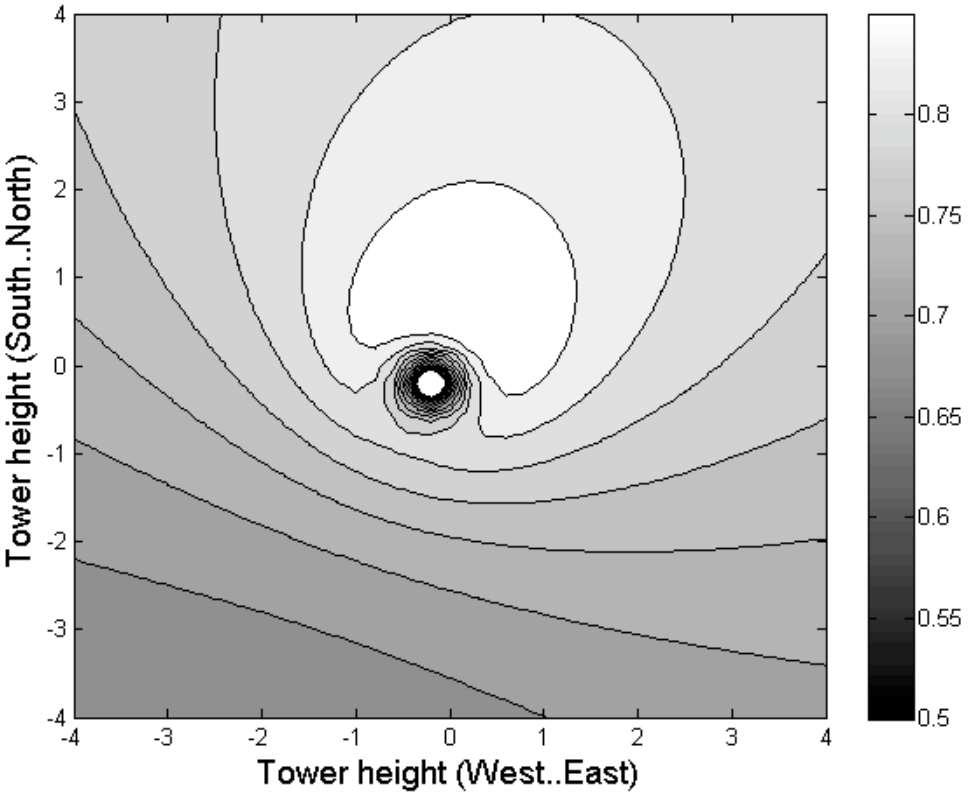


Figure 7: Plot of the cosine and incidence angle losses associated with the area surrounding the tower receiver. Solar position is 18.5 deg zenith and 250 deg azimuth and the tower is located at point [0,0].

Since the solar position is in the southwestern sky (originating towards the lower left-hand corner of the figure), the area exhibiting the highest efficiency is to the upper right of the tower, which is located at point [0,0] on the plot. This prediction is confirmed with observation of the figure. It is important to note the effect of the reflected radiation incident angle on the efficiency of each position on the map. As the heliostats are placed nearer to the tower base, the path of the reflected radiation causes a high incidence angle, and the efficiency decreases significantly. This efficiency reduction is evident with a closer look at the area immediately surrounding the receiver, as shown in Figure 8, which is a magnified view of the field shown in Figure 7. Each contour line corresponds to a 2.5% change in reflected radiation (also note that the white space immediately surrounding the tower position indicates efficiency that is less than the minimum value of 50% selected for this figure).

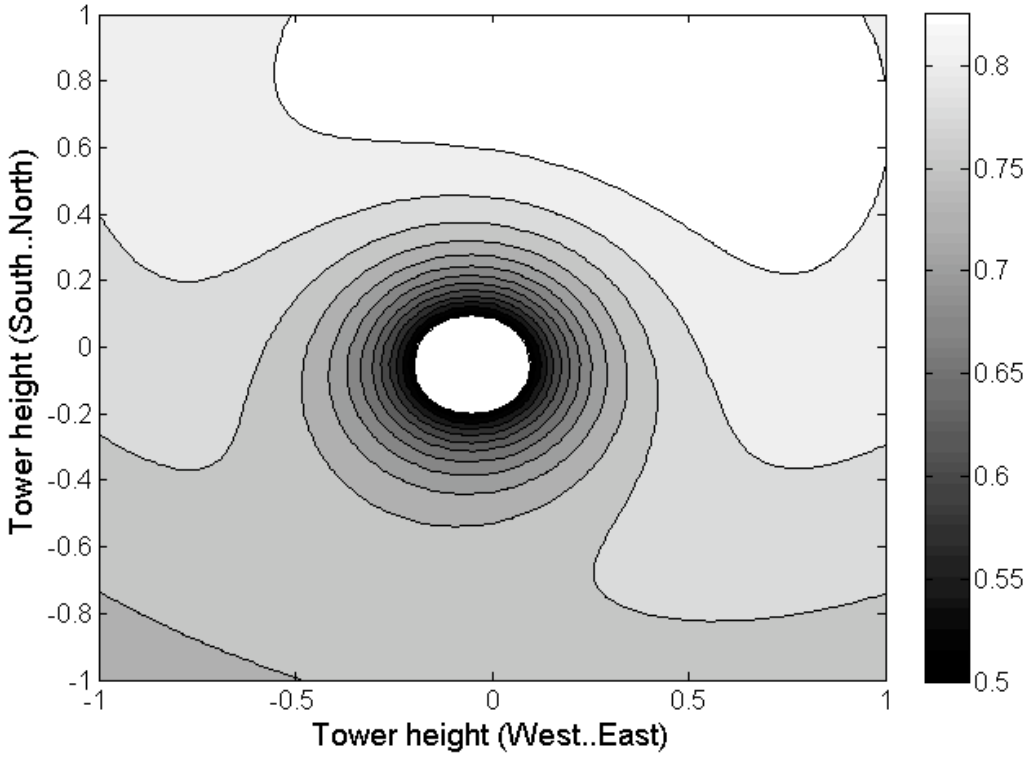


Figure 8: A magnified view of the field efficiency in the area surrounding the tower receiver.

Atmospheric attenuation

The third field loss is due to atmospheric attenuation, or the scattering of beam radiation as the reflected heliostat image passes through the air towards the receiver. This loss is a function of distance of the heliostat from the receiver surface, ambient humidity, and plant geographical elevation. To illustrate typical losses associated with atmospheric attenuation, the relationship for the Solar II location is presented in Figure 9 for visibilities of 23 *km* and 5 *km*. Although this correlation was developed for a specific location, the dependence on elevation is only slight compared to the dependence on visibility. Therefore, the correlations presented in Eq. (2.6) and Eq. (2.7) (Hottel, 1976) and illustrated in Figure 9 can be safely used for dry climates until more detailed site-specific information becomes available.

Visibility = 23 km:

$$Loss(\%) = 0.6739 + 10.46R - 1.70R^2 + 0.2845R^3 \qquad (2.6)$$

Visibility = 5 km:

$$Loss(\%) = 1.293 + 27.48R - 3.394R^2 \tag{2.7}$$

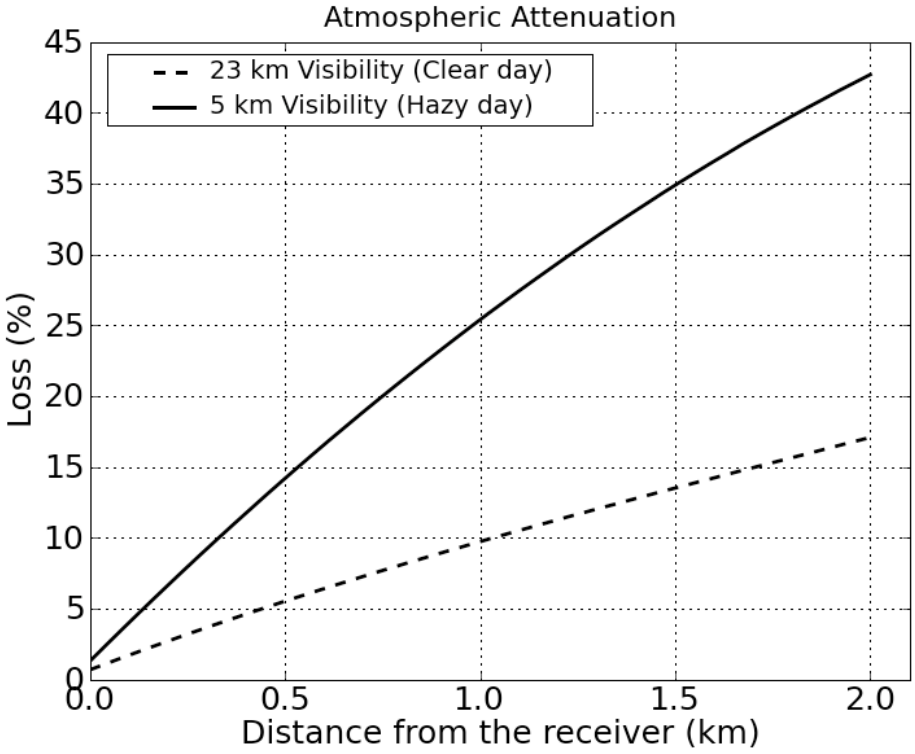


Figure 9: The atmospheric attenuation for Daggett, CA, at visibilities of 23km and 5km (Hottel, 1976).

Spillage Losses

The final loss that is a function of the heliostat's position relative to the tower is in reality the cumulative result of several factors. The spillage, or proportion of the reflected image that fails to intercept the receiver surface is a function of tracking precision, surface uniformity, sway movement of the tower, the shape of the sun (the sun acts as a disc rather than a point in providing beam radiation), environmental factors like wind velocity and heliostat foundation settling, and the controller and algorithm issues identified by Stone and Jones (1999). These factors contribute to a reflected image at the receiver surface that is errant from its intended path to the receiver and distorted from its shape as it originates from an individual heliostat. The image inflation and deviation can be

modeled using a probability distribution centered on the heliostat surface reflection vector ($\hat{R}$) with a standard deviation of between 1 and 3 *mRad* (Kistler, 1986) for the non-sunshape considerations, depending on the construction of the heliostat.

The total image aberration including the errors and the sunshape correction is illustrated in Figure 10. Calculation of the proportion of the radiation that is incident on the receiver surface is also dependent on complex aiming schemes that are discussed briefly later in this section. For more information on aiming calculations, see Kistler (1986) or Walzel, et. al (1977).

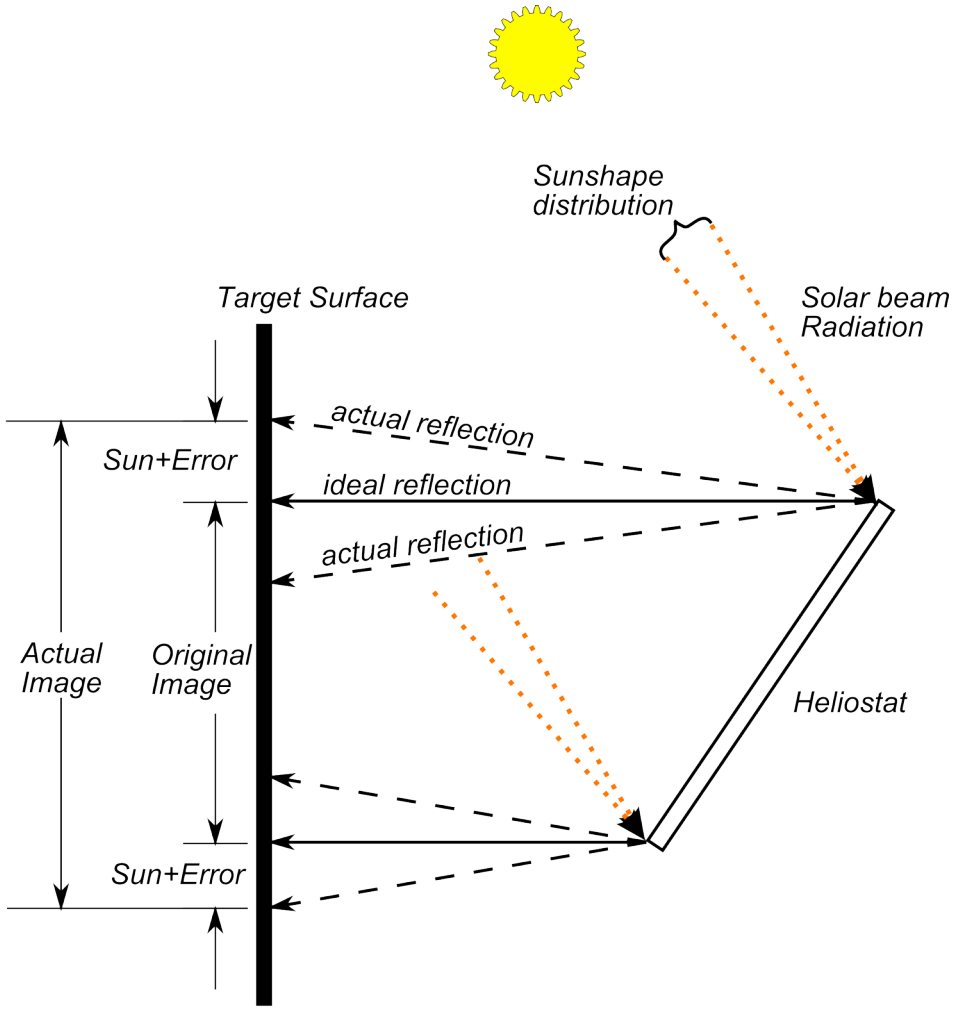


Figure 10: Image aberration for the heliostat as a function of errors and the sun shape. The resulting reflected image on average is larger than the ideal reflected image.

The previously discussed losses can be attributed to the position of the heliostat in the field relative to the central tower, tracking error, or losses associated with the construction of the heliostats. The two remaining losses – shading and blocking – are exclusively a function of the heliostat's placement with respect to the other heliostats in the field.

Shading and Blocking Effects

Shading occurs when one or more heliostats in the field is partially obscured from incoming solar radiation by a neighboring heliostat, and blocking occurs when the reflected image from a heliostat is obstructed from reaching the receiver by an adjacent heliostat. These phenomena are illustrated in Figure 11, as presented in Stine and Geyer (2001).

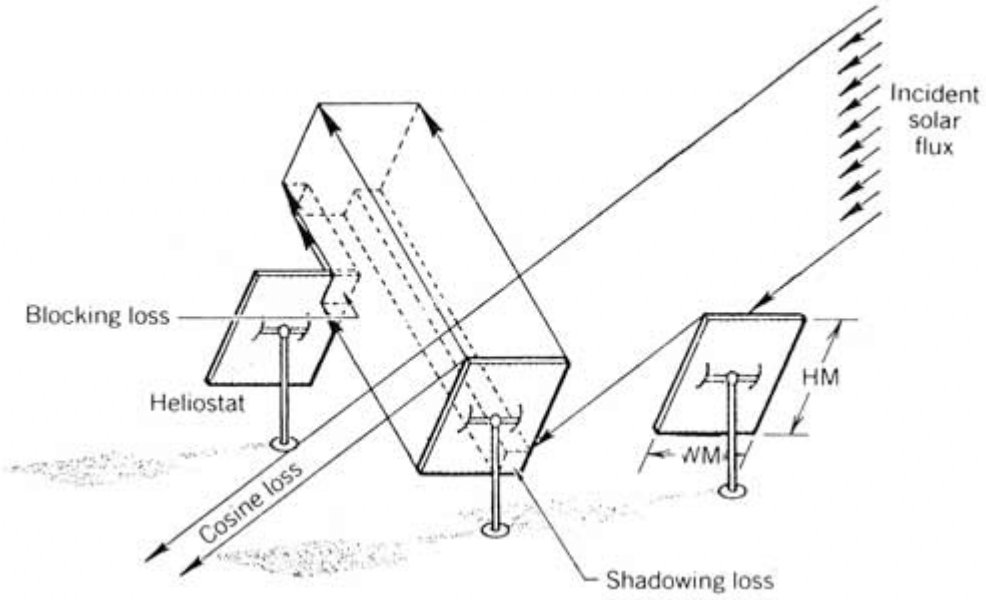


Figure 11: Heliostat blocking and shading losses (Stine and Geyer, 2001).

Losses due to shading and blocking can be managed with an adequate field layout scheme, but the elimination of shading and blocking must be tempered with the other heliostat losses such that an optimal design arises. A number of codes have been developed to handle this task, or to compare existing designs. These include the University of Houston code (Lipps and Vant Hull, 1977), MIRVAL (Leary and Hankins,

1979), DELSOL3 (Kistler 1986), HFLCAL (Kiera and Schiel, 1989), SolTRACE (Wendelin, 1989), and the Heliostat Growth Method (Sanchez and Romero, 2006). The DELSOL3 code is utilized in this research.

2.1.2 Heliostat Field Layout

Heliostat field design has been an area of interest in past research efforts and the capability of effectively analyzing their design and performance is increasingly becoming more tractable as computing power has increased. While a radially staggered method for heliostat field layout has traditionally been employed, more computationally demanding iterative methods have also been proposed (Sanchez & Romero, 2006). Figure 12 below illustrates the staggered heliostat method that is used in this research.

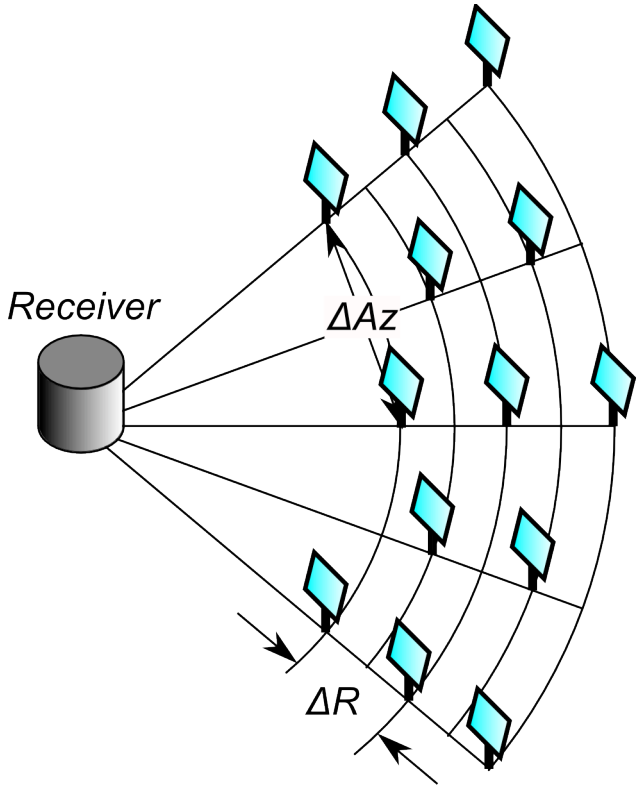


Figure 12: Radially staggered configuration. ΔR represents the radial distance, and ΔAz represents the azimuthal distance between heliostats at a given radius.

The radially staggered heliostat field layout method in plant simulation requires minimal use of computing resources and provides an established and well-tested methodology for

designing and optimizing the heliostat field layout. Additionally, the established capability of Fortran codes such as RCELL, DELSOL3, and SOLERGY to provide an optimized relationship between tower height, receiver dimensions, and field layout introduces additional incentive to use this method (Sanchez and Romero, 2006). The limitations to this approach stem from the "fixed" nature of the heliostat grid. After the placement of the first heliostat relative to the central receiver has occurred, the placement of each additional heliostat may not be at the optimum point to minimize blocking, shadowing, and other effects.

For the method used in this research, the optimizing code (DELSOL3) uses radial and azimuthal spacing correlations that vary with position in the field depending on the loft angle (θ_L) between the heliostat, the ground, and the tower, and are determined using curve fits to optimized layouts reported by the University of Houston. Several optimization curve fit options are available, but the default is given by Eqns (2.8) and (2.9) (Stine and Geyer, 2001) where ΔR and ΔAz are the distance between heliostats in the radial and azimuthal directions, respectively, as illustrated in Figure 12 above.

$$\Delta R = \left(1.1442 \cdot \cot\theta_L - 1.0935 + 3.0684\theta_L^2\right) H_{helio} \tag{2.8}$$

$$\Delta Az = \left(1.791 + .6396 \cdot \theta_L\right) \cdot W_{helio} + \frac{0.02873}{\theta_L - 0.04902} \tag{2.9}$$

where:

$$\theta_L = \frac{\pi}{2} - \theta_t$$

θ_t = the angle between a vector from the heliostat to the tower and vertical

H_{helio} = the height of the heliostat mirror

W_{helio} = the width of the heliostat mirror

To illustrate the azimuthal and radial spacing, Figure 13 is included which shows these relationships plotted for a 100 m tower with $10 \times 10\,m$ heliostats in the field. The process of placing heliostats within the heliostat field layout is performed by DELSOL3, and

more detail on the program and its implementation in this research is provided in Chapter 5.

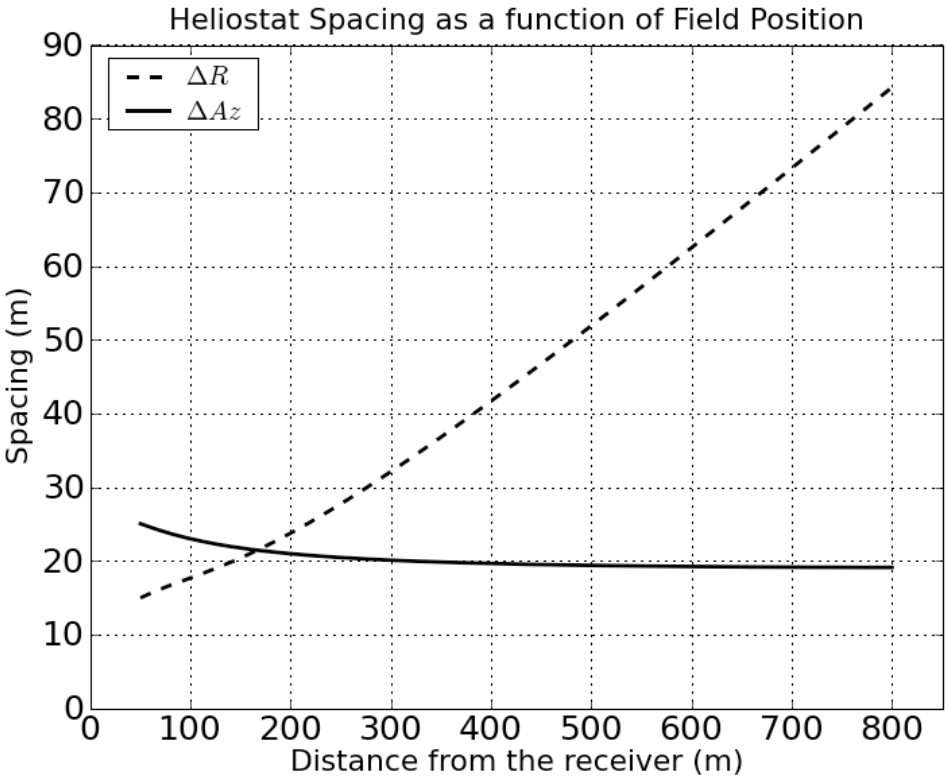


Figure 13: The radial and azimuthal heliostat spacing for a 100m tower with 10x10m heliostats using the relationships defined above.

The basis of the correlation developed in Eqns (2.8) and (2.9) is that the heliostat field will completely surround the receiver tower on all sides. This type of field, called the *surround field*, typically accompanies a cylindrical receiver that has an absorber surface exposed around its circumference. As previously mentioned, the field is generally weighted such that the majority of the heliostats are to the north of the tower (assuming the field is in the northern hemisphere). This was the case for the Solar II plant in Daggett, CA, as shown in Figure 14 (Google).

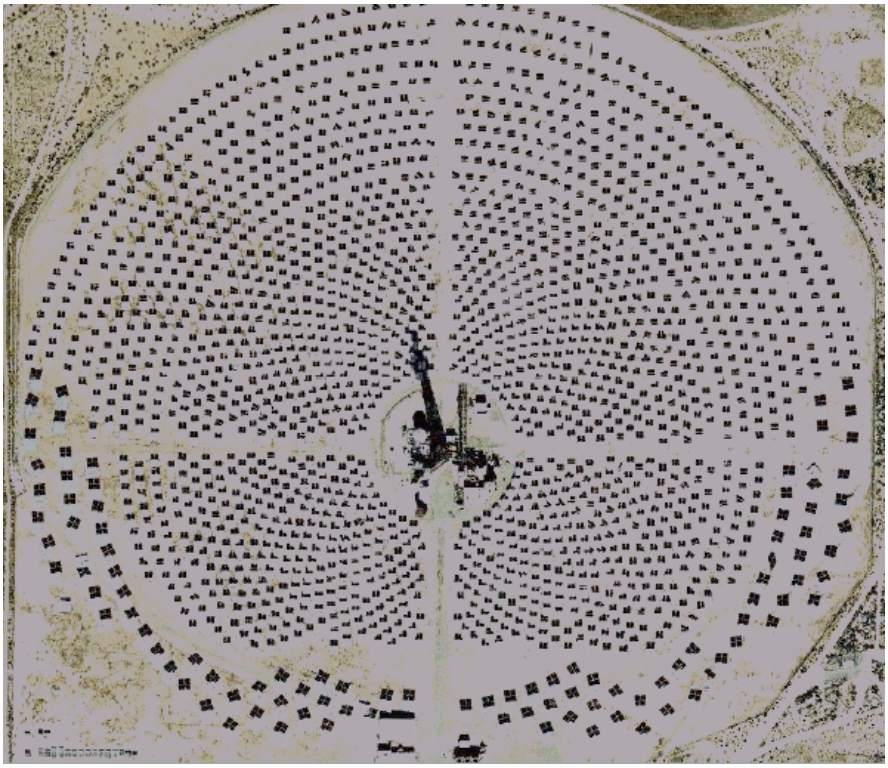


Figure 14: An aerial view of the Solar II plant (Google Maps).

A second possible configuration is a field that is exclusively to the north of the tower. With the heliostat field relegated to the northern side of the tower, the back of the tower can be insulated and the receiver is often placed inside a cavity to reduce thermal losses. This configuration requires its own set of spacing correlations, as given in Eqns (2.10) and (2.11) (Kistler, 1986).

$$\Delta R = \begin{bmatrix} 63.0093 - 0.587313\theta_L + 0.0184239\theta_L^2 \\ + \cos\phi \left(2.80873 - 0.14805\theta_L + 0.0014892\theta_L^2 \right) \end{bmatrix} H_{helio} \tag{2.10}$$

$$\Delta Az = \begin{bmatrix} 2.46812 - 0.040105\theta_L + 9.2359 \cdot 10^{-4}\theta_L^2 \\ + \cos\phi \left(0.17345 - 0.009113 + 1.2761 \cdot 10^{-4}\theta_L^2 \right) \end{bmatrix} W_{helio} \tag{2.11}$$

where:

ϕ = heliostat azimuth angle, 0° is south.

The layout provided by these correlations should be treated as a starting point, and optimization of the heliostat spacing is desirable.

2.1.3 Aiming Techniques

Once the optimized heliostat field layout has been determined, the efficiency of each heliostat in the field can be determined, and the average of these efficiencies at a given point in time gives the net field efficiency. Efficiency in this case is defined in Eq.(2.12) as the ratio of the power that successfully reaches the receiver surface to the product of the incident beam-normal radiation and the total surface area of the mirrors in the heliostat field.

$$\eta_{field} = \frac{\dot{Q}_{inc}}{I_{bn} \cdot A_{helio} \cdot N_{helio}} \tag{2.12}$$

where:

A_{helio} = the area of a single heliostat

N_{helio} = the total number of heliostats in the field

The shape of the field also determines the flux distribution on the receiver surface, and a number of aiming techniques can be used to control how the flux is distributed to maximize performance while avoiding situations that can exceed the receiver's flux limit. Two common techniques for the external cylindrical receiver include the simplest scheme, which focuses all of the heliostat images at a single target point at the receiver azimuth angle corresponding to the azimuth angle of the heliostat in the field (Kistler, 1986). This is shown in Figure 15, where the cylindrical receiver surface has been "unwrapped" and laid flat on a plane, and the corresponding azimuth angle of the receiver is shown. A second common aiming scheme distributes the heliostat images over the vertical axis as well until the spillage loss begins to increase. The inner heliostat images that are easier to aim can be positioned near the top and bottom of the receiver surface to more evenly distribute the flux, while the larger heliostat images are relegated to the central portion of the receiver (Kistler, 1986). This concept is illustrated in Figure 15.

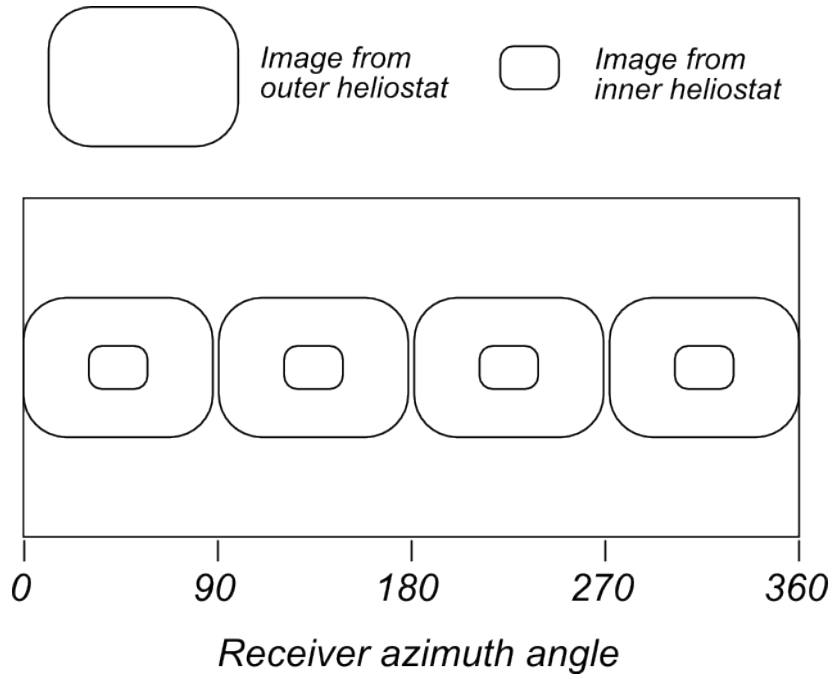


Figure 15: A flux distribution scheme where the heliostats are aimed at a single point according to their azimuth angle in the field.

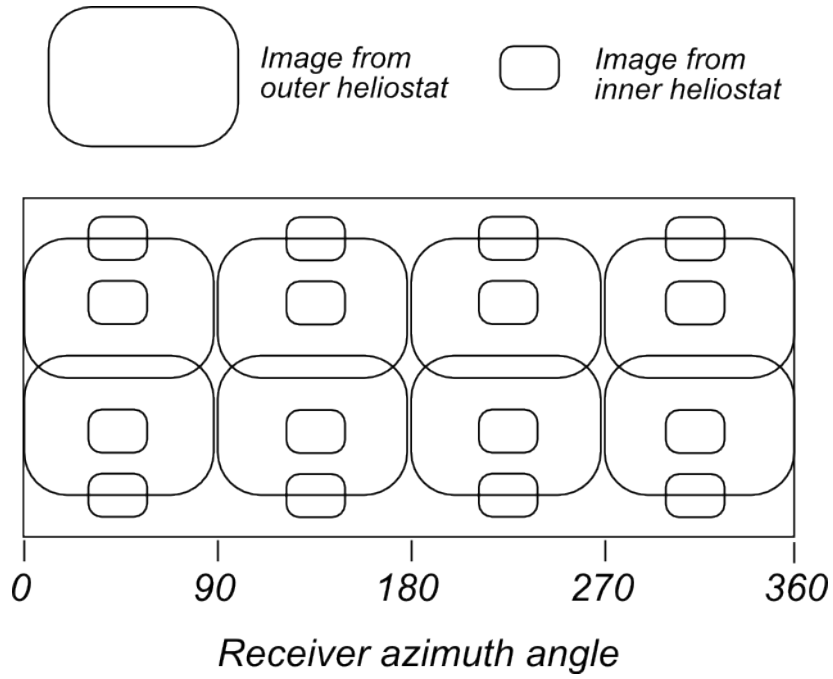


Figure 16: A flux distribution scheme using a "smart aiming" technique, where images are distributed over the height of the receiver.

The combination of the total receiver field efficiency and the flux distribution on the receiver are the defining characteristics of a heliostat field, and the field can be fully implemented into a system model using these two pieces of information. This information can be produced by a field design code such as DELSOL3 or the others mentioned previously. DELSOL3 was used for this research, and the following section discusses the implementation of this information into the model.

2.2 The Heliostat Component Model in TRNSYS

With the information presented in the previous section – the total heliostat field efficiency and the flux distribution on the receiver, both as a function of solar position – the heliostat field performance is fully defined. Additional parasitic losses associated with heliostat tracking and startup/shutdown should be accounted for, but they are peripheral to the core behavior of the field. Implementation of this data in a system model in TRNSYS is best served by supplying the field efficiency as a function of the solar position with a special heliostat field component. The other parasitic losses can also be tracked with a dedicated component.

Chapter 1 introduced an existing TRNSYS heliostat field model (Type 394), and noted that several shortcomings in the model prevented it from being useful in the current research. The heliostat component was re-written to rectify these issues and to cater more directly to the model developed for this research. The programming inconsistency that caused the field control to be neglected was corrected, and both the DNI input and the average heliostat reflectivity parameter were removed to reduce confusion and possible double-accounting. The resulting component still includes the option to account for wind velocity and parasitic losses. The field efficiency is determined by interpolating a two-dimensional array containing the efficiency values over a range of solar azimuth and zenith angle positions. The number of solar azimuth and zenith angles used to create this array must be provided. Table 9 summarizes the parameters, inputs, and outputs for the heliostat field component (Type 221) that was written during this research, and the

units and default values are also indicated. The required format of the file used to provide the heliostat field efficiency is described in more detail in Section 5.2.4 of the *Plant Sizing and Optimization* chapter.

Table 9: A summary of the parameters, inputs, and outputs for Type 221, the heliostat field component

Parameters *[default | units]*			**Inputs** *[default | units]*			**Outputs** *[units]*	
Unit no of input file	10	-	Wind speed	0	m/s	Parasitic tracking power	kW-hr
No of zenith angle data points	8	-	Defocus factor	1	-	concentrator field efficiency	-
No of azimuth angle data points	12	-	solar zenith angle	90	deg		
Number of heliostats	0	-	solar azimuth angle	0	deg		
Startup energy of unit	0	J					
Power to track 1 unit	0	kW-hr					
Max allowed wind-speed	999	m/s					

The receiver flux distribution is not implemented using the heliostat field component model; instead it is applied directly using the receiver model since the receiver component makes use of the information contained in the flux distribution. The receiver component also makes use of the DNI by multiplying the flux distribution by the instantaneous insolation value. The flux distribution is covered in detail in Section 5.2 of the *Plant Sizing and Optimization* chapter, and also in the *Central Receiver Model* chapter (Chapter 3).

3 Central Receiver Model

Concentrating Solar Power (CSP) technologies operate on the premise that thermal power can be collected more efficiently and cost-effectively at higher flux concentrations. The *Power Tower* technology takes advantage of this principle, operating at radiation flux concentrations *600-1200* times normal terrestrial direct-normal radiation levels. This high of a flux concentration creates significant design and operating challenges to assure that the concentrated solar flux is evenly distributed over the absorber surface to avoid thermal failure. The flux concentration varies with solar radiation availability, cloud transients, solar position, and plant configuration. To further complicate matters, the receiver surface is often directly exposed to ambient conditions, with varying wind-speed, atmospheric pressure, and temperature as weather changes.

To better understand the behavior of the receiver under these conditions, a detailed thermal model is developed. For operating rigor, the receiver model must be capable of resolving the highly concentrated solar flux distribution over the absorber surface to assure that its design flux limits are not exceeded. The receiver model also considers losses due to convection, radiation, and reflectance. Previous work in this area has not yielded a detailed, fundamentals-based receiver model that is suitable for use in long-term energy simulations. Therefore, an effort was made to develop a model of this type and incorporate it into TRNSYS.

The following chapter discusses the thermal modeling process for an external, cylindrical-type receiver. This process involved initial model construction and debugging in EES followed by translation of the model into Fortran for use in TRNSYS. Presentation of the receiver model details is prefaced by a more general overview of the technology, including system geometry, operating parameters, and a brief review of molten salt working fluids.

3.1 Central Receiver Model Background

The central receiver design has traditionally been one of two possible configurations – either a fully exposed cylindrical surface, or a concave surface nested inside of a protective cavity. The exposed cylindrical surface, commonly referred to as the *external receiver*, consists of a number of individual receiver panels arranged in a vertical cylinder atop a tower. These panels are typically exposed to ambient conditions without glazing or protection. The *cavity receiver*, like the external receiver, is often an assembly of multiple panels. However, this configuration provides some degree of protection from the ambient conditions, as it is situated inside an open cavity which reduces radiation and convection losses.

Both configurations have their merits and drawbacks, and either might be better suited for an application once cost and location-specific variables have been factored in. For example, while the *external receiver* sacrifices thermal efficiency in increased exposure to its surroundings, it can accommodate a larger heliostat field circumferentially surrounding the receiver. The cavity receiver gets a boost in efficiency with its increased isolation from ambient conditions, but its geometry restricts the layout of the field to a portion of the azimuthal angles. Figure 17 shows examples of the cavity and external receiver configurations.

The solar receiver geometry to be employed in this model is based on the Solar II demonstration project, which was conducted from June 1996 to April 1999 near Barstow, CA (Pacheco, et al., 2002). The receiver at Solar II successfully demonstrated the power tower technology using an external receiver, and provides a sound conceptual basis for formulating this model.

Figure 17: Two possible receiver geometries include the external receiver (LEFT) and the cavity receiver (RIGHT), among others. The external receiver is modeled in this report.

3.1.1 Receiver Geometry

The cylindrical surface of the receiver is formed with a number of smaller rectangular panels, each consisting of a number of vertical tubes filled with heat transfer fluid that run in parallel between a common lower and upper flow header. The tubes can vary in size from 20*mm* (outer diameter) to 45*mm* (Lata, et. al, 2006) and are generally stainless-steel alloy. The number of tubes per panel should be chosen such that heat absorption is optimized with respect to thermal losses and the pressure head across the receiver. Additionally, these concerns are subject to other considerations such as receiver diameter, the length of the flow path through the panels, and materials. The tube diameter and number of panels comprise two of the several important parameters for optimization in the modeling process. A single panel that has been removed from the receiver tower is shown in Figure 18.

Figure 18: Solar Two Receiver panel. The receiver surface is shown facing upwards with the back-side insulation underneath.

To maximize absorption of incident solar flux, the tubes are coated with high-absorptivity paint such as the black-matte Pyromark™ paint used for Solar II. The average absorptivity of this particular coating is claimed to be between 94% and 95% over all thermal radiation wavelengths (Pacheco, J.E., et al., 2002) and the coating is moderately selective with an emissivity of approximately 0.88.

Heat transfer fluid flow configurations can vary from receiver design to design, depending on material thermal strain constraints, heliostat field distribution, and pumping requirements. Figure 19 shows plan views for eight of the many possible flow configurations for the external receiver. Multiple flow-paths for the heat transfer fluid can be taken simultaneously, and this is illustrated in designs *1-4* in Figure 19.

In the Solar II receiver, the working fluid (heat transfer fluid) follows the serpentine series path through the receiver panels as shown in Figure 20 below. The heat transfer fluid enters on the north-most panels and proceeds in series through half of the panels before exiting from the two south-most panels of the receiver. Figure 20 also illustrates the flow path between adjacent panels.

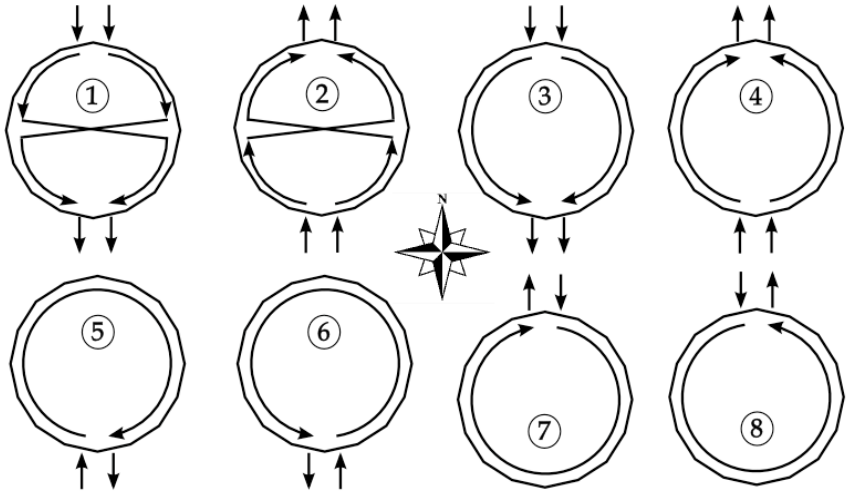


Figure 19: Eight possible flow configurations for the external receiver. Each configuration is presented from a top-down, axial line of sight viewpoint.

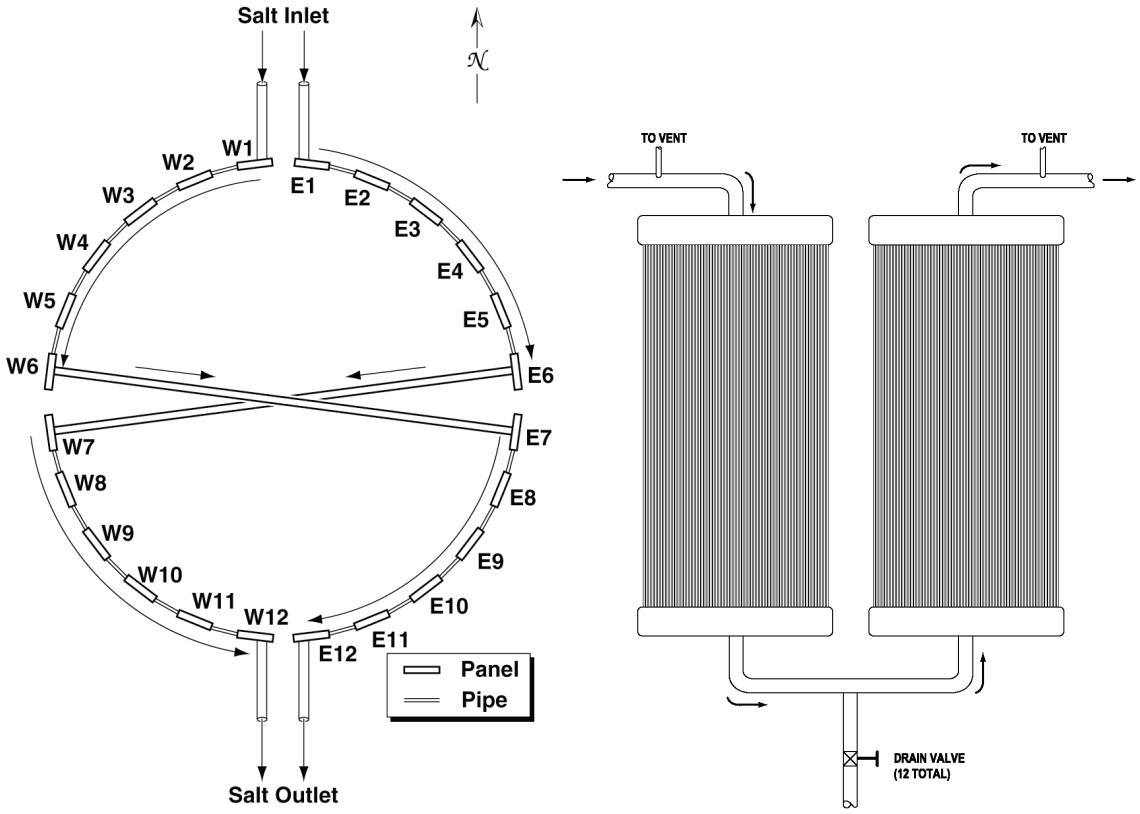


Figure 20: The Solar II cross-over flow pattern, as viewed from the top (LEFT). The flow arrangement between two adjacent receiver panels (RIGHT) (Reilly and Kolb, 2001).

3.1.2 Operating Parameters

Due to the large radiation flux concentration on the receiver surface, the receiver is able to generate much higher heat transfer fluid temperatures than in the solar parabolic trough counterpart. In receivers using a molten salt as the heat transfer fluid, maximum temperatures are constrained by receiver materials of construction and in some cases by the heat transfer fluid stability or boiling point.

For the receiver in the Solar II system, the heat transfer fluid (a molten nitrate salt consisting of 60% $NaNO_3$ and 40% KNO_3) was pumped from the 'cold' storage tank at 288°C (550°F, 561 K) into the receiver, where it was heated to 565°C (1050°F, 838 K) and pumped to the hot storage tank (Reilly and Kolb, 2001). The mass flow rate of the heat transfer fluid through the receiver is modulated to match the desired outlet temperature from the tower as the incident flux varies throughout operation.

Flux distribution on the receiver is managed by controlling the position of the individual heliostats in the field with temperature-sensing feedback including thermocouples on the internal tube surfaces and optical measurements (Pacheco, 2002). Aiming schemes for the heliostats attempt to produce a flux distribution on the receiver that minimizes spillage while avoiding unnecessarily dense or sparse concentration at any location. In the case that the flux on an area of the receiver is insufficient, the molten salt heat transfer fluid is at risk of freezing in the receiver tubes. This problem occurred in the Solar II demonstration project and required the installation of a heat trace system (Litwin, 2002). Conversely, a flux that is too high could lead to 'burnout' of the receiver tubes, or cause premature failure of the receiver panels due to cyclic thermal stress. Any of these circumstances could potentially be severely damaging to the equipment, and should be avoided. The flux limit for Solar One was maintained at 600 kW/m^2, while Solar II was able to sustain flux limits of 850 kW/m^2 (Pacheco, 2002).

As an example of a flux distribution on an external receiver, Figure 21 shows a modeled flux distribution at a solar azimuth and zenith of 230° and 18°, respectively. To understand the meaning of the plot, it should be noted that the cylindrical surface of the

external receiver has been mapped to a planar surface in which the progression from *0°* to *360°* along the horizontal axis corresponds to the progression around the circumference of the receiver. The maximum flux occurs at the northernmost panels where the angular position is *180°*. The vertical position of the cylinder has been normalized to the height of the receiver. The bottom of the receiver falls at a normalized vertical position of zero, and the top falls at one.

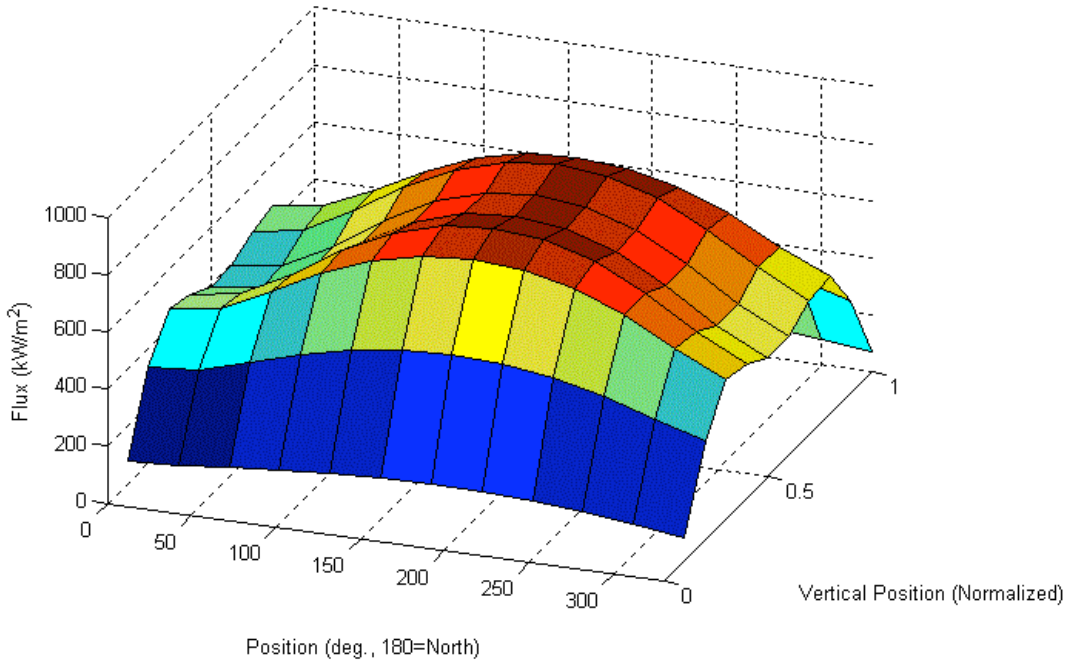


Figure 21: The flux distribution on an external receiver at 35°N latitude at 1:00pm solar time on August 6th. The circumferential position is defined such that 180° faces north and 270° faces east.

3.1.3 Heat Transfer Fluids

While the use of molten nitrate salt as a heat transfer fluid was successfully demonstrated in the Solar II project, it has not been universally adopted for use in current power tower development. Some drawbacks including problems with salt freezing and cost hinder the use of molten salt in some circumstances. Coolants such as steam (Romero, et. al, 2002) and air (Kribus, et. al, 2001) have been successfully demonstrated and are an alternative to molten salts for the power tower technology.

Since by definition, the term *salt* refers to any chemical compound that is formed through the combination of an acid and a base (Moore, et al., 2002), the molten salt class of substances is broad and can include many different molecular groupings. Therefore, the properties of molten salts can vary greatly from substance to substance. Often, molten salts are mixed with other salts or compounds to achieve desired properties, and significant effort is underway in this area to improve the performance of heat transfer fluids for solar applications (DOE, 2008). Challenges for molten salts remain, including a relatively high freezing temperature ranging from 530*K* (257°C, 494°F) for the 60% $NaNO_3$, 40% KNO_3 mix to 775*K* (502°C, 935°F) for a salt like 58% KF, 42% ZrF_4 (Williams, 2006).

Of tremendous importance to the modeling process is an accurate and complete set of data characterizing the receiver heat transfer fluid. Through collaboration with the nuclear reactor research group at the University of Wisconsin - Madison (Sridharan, 2007), data on eleven potential heat transfer fluid salts were obtained. The implementation of molten salt in the nuclear industry as a high-temperature heat transfer fluid introduces many of the same benefits and challenges as in concentrating solar power applications (Forsberg, 2007).

To provide the reader with a sense of the behavior of the molten salt used in Solar II, plots of the thermal conductivity, density, specific heat capacity, and viscosity are included below (Williams, 2006). These properties are implemented in the central receiver thermal model and have also been included in the EES property data.

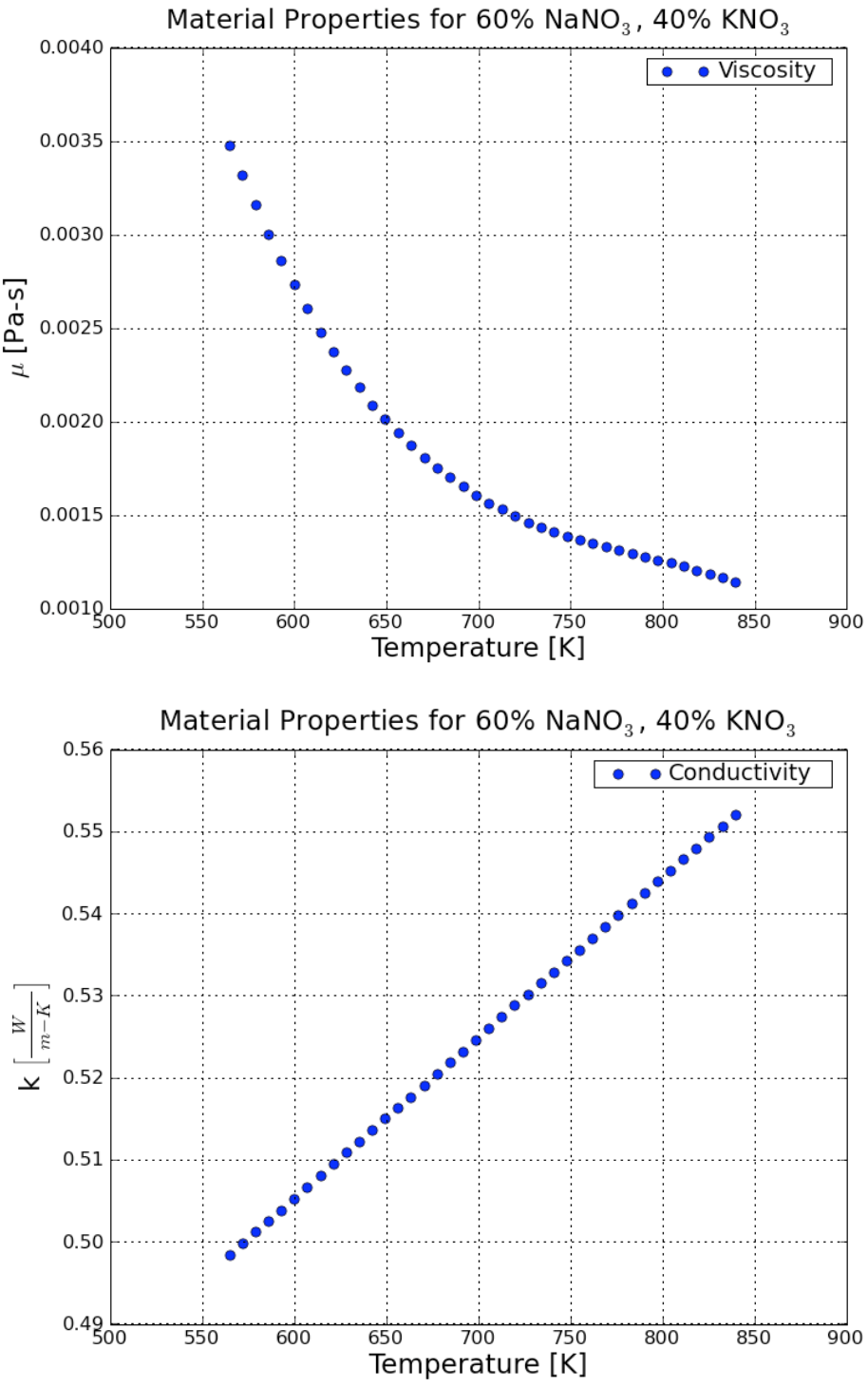


Figure 22: The viscosity and conductivity of NaNO3 + KNO3 over its useful temperature range

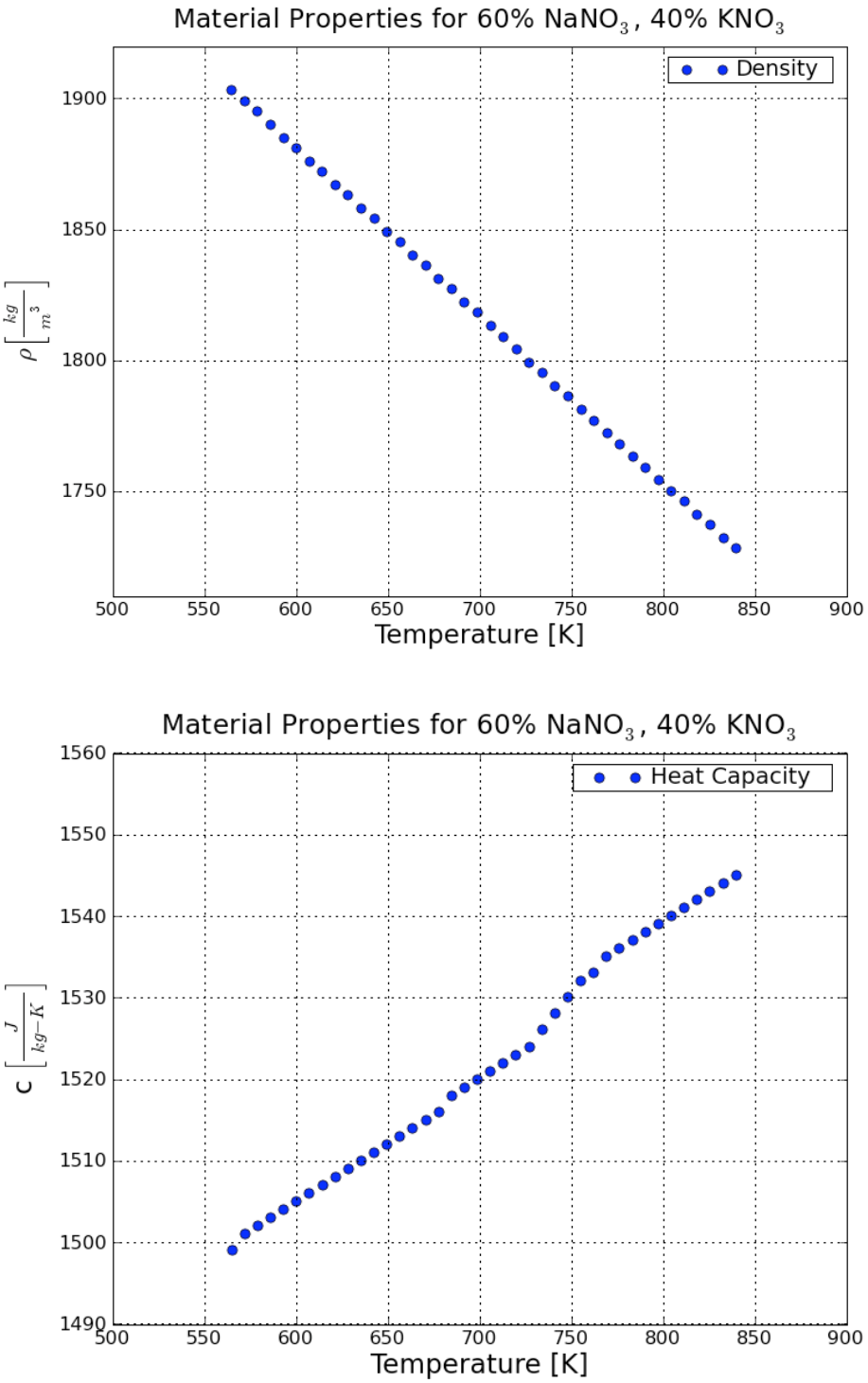


Figure 23: Density and specific heat of $NaNO_3$+KNO_3 as a function of temperature

3.2 Receiver Thermal Model in EES

The purpose for developing a detailed receiver model is multifold. First, the goal of a detailed thermal model is to accurately predict the net thermal power absorbed by the receiver. This means that both the incident absorbed flux be predicted as well as the thermal losses. Second, the model must be capable of predicting the flux distribution to assure that the flux limits established for the materials of construction and heat transfer fluid design are not exceeded. Lastly, a detailed receiver model allows alternative fluid circuiting arrangements to be considered and their individual performance evaluated. The developed model calculates conditions at multiple points on the receiver which allows more accurate estimates of temperature distributions to be made; thereby, allowing the user to model a receiver that is geometrically unique from previously modeled systems.

3.2.1 Energy Balance and Model Formulation

The basic building block of the receiver model is a single tube of length Δx, where Δx is some finite portion of the overall panel length in the vertical direction. This element is subject to multiple heat transfer mechanisms, including incident radiation ($\dot{q}_{inc}$), external convection ($\dot{q}_{conv}$), and radiation exchange with the surroundings ($\dot{q}_{rad}$). Radiation that is simply reflected from the tube surface ($\dot{q}_{ref}$) is an additional consideration. The energy balance for each element is presented in Figure 24.

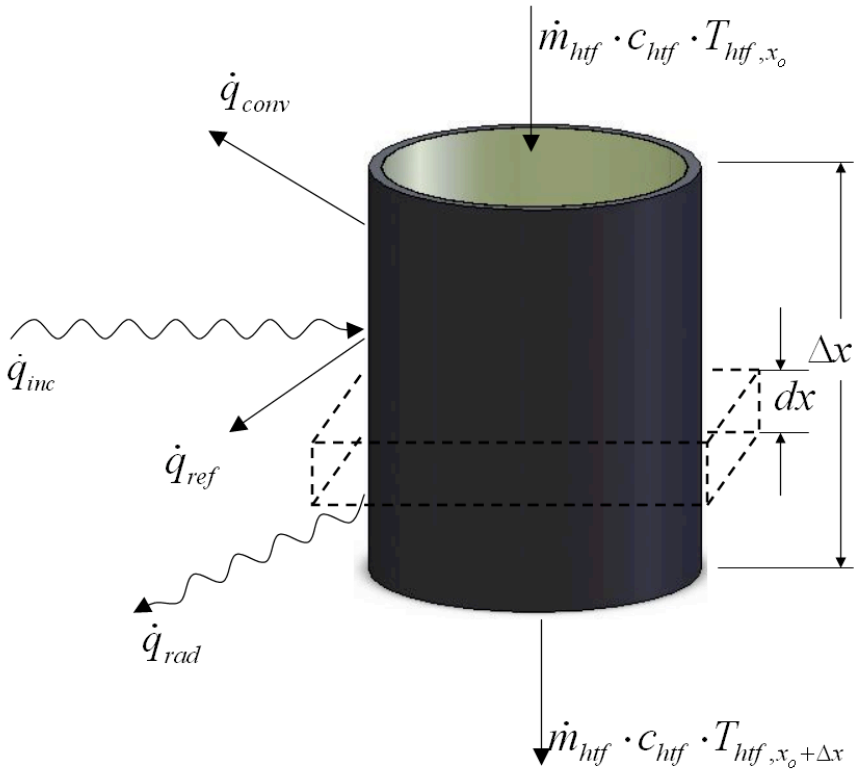


Figure 24: Energy balance on a receiver tube element.

Because the resolution of the flux data on the receiver surface provided by the DELSOL3 modeling tool is limited to approximately one data point per panel in the circumferential direction and several flux points per panel in the vertical direction, each of the parallel tube elements at vertical position *x* in a receiver panel are represented by the same energy balance. Therefore, the result of an energy balance applied to a single tube at position *x* on panel *N* can be scaled by the number of tubes in that panel. Since each tube is then essentially a facsimile of its neighbor, tube-to-tube conduction and radiation exchange is neglected. Axial conduction is also neglected since the much larger internal convection due to salt flowing in the tubes dominates over the relatively large resistance to conduction.

The various heat transfer flows can be expressed more specifically in terms of a differential element of length *dx*, also shown in Figure 24. For each differential element

dx, the overall steady-state energy balance on the heat flow components leaving and entering the control volume is:

$$\dot{q}_{fluid} = \dot{q}_{inc} - \left(\dot{q}_{ref} + \dot{q}_{rad} + \dot{q}_{conv}\right) \tag{3.1}$$

More specifically, these terms can be expressed as integrals with respect to axial position, *x,* over the length of the element, *Δx.* The incident irradiative flux on each panel with one azimuthal data point and multiple vertical data points is given by:

$$\begin{aligned} \dot{q}_{inc}(x) &= D_{tube} n_t \cdot \int_{x_o}^{x_o+\Delta x} P''_{field}(x)\,dx \\ \dot{q}_{inc,x} &= P''_{field} \cdot \Delta x \cdot D_{tube} \cdot n_t \end{aligned} \tag{3.2}$$

where P''_{field} is determined using the flux distribution shown in Figure 21, and n_t is the number of tubes in each panel.

The energy that is initially reflected from the tower is represented by the $\dot{q}_{ref}$ term. The receiver model assumes a constant, spectrally independent, hemispherical absorptivity (α) for the tower surface. Since the tower surface is opaque, the reflectivity is *1-α.*

$$\begin{aligned} \dot{q}_{ref}(x) &= (1-\alpha) \cdot D_{tube} \cdot n_t \int_{x_o}^{x_o+\Delta x} P''_{field}(x) \cdot dx \\ \dot{q}_{ref,x} &= (1-\alpha) \cdot D_{tube} \cdot n_t \cdot P''_{field} \cdot \Delta x \end{aligned} \tag{3.3}$$

Radiation emitted from the receiver to the ambient surroundings is considered through the $q_{rad,i}$ term. A view factor from the tower to the surroundings ($F_{t,s}$) was calculated using the EES correlation for a cylinder that is surrounded on both sides by parallel cylinders of the same diameter, all of which lie in a single plane. This arrangement is shown in Figure 25.

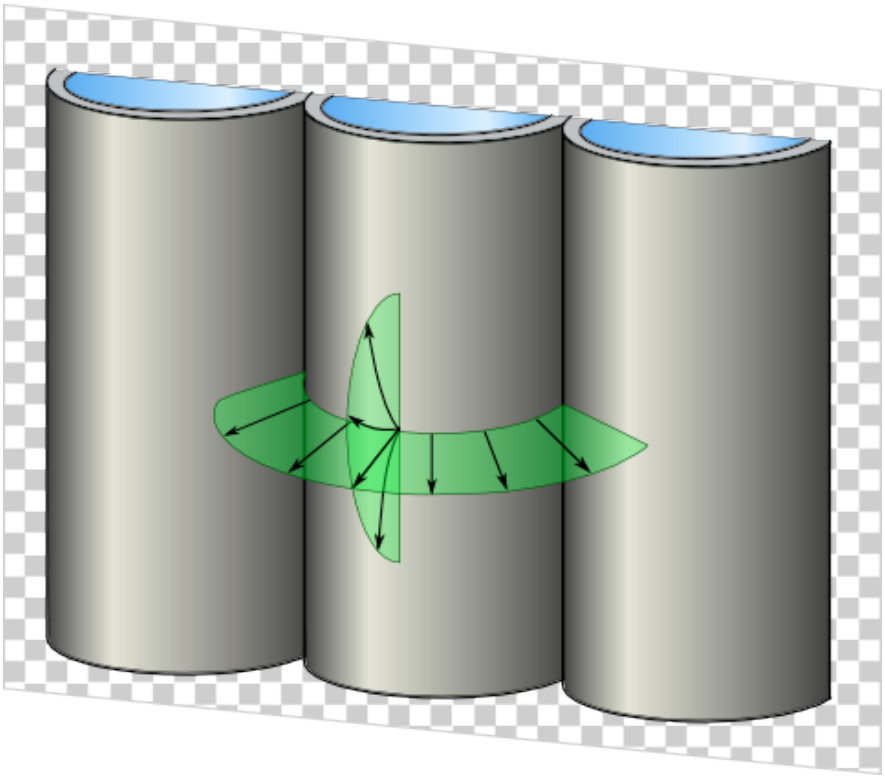


Figure 25: An illustration of the view factor seen by a tubes lying in a plane. The view factor is the range of the surroundings visible to the tube.

The value of the view factor (0.6366) adjusts the outbound radiation from the half-cylinder exposed to the surroundings to account for radiation exchange between the cylinders. Because the cylinders are assumed to be very close or touching each other, the view factor scales the effective area of the tube to be the projected area of the tube (the outer diameter times the tube height).

Emissivity, ε , is given as 0.88 (Taumoefolau, 2004). Thus the rate of energy transfer from the panel by radiation is:

$$\dot{q}_{rad}(x) = \sigma\varepsilon\pi \cdot \frac{D_{tube}}{2} \cdot F_{t,s} \cdot n_t \int_{x_o}^{x_o+\Delta x} \left(T_s^4(x) - T_{amb}^4\right) dx$$
$$\dot{q}_{rad,x} = \sigma\varepsilon\pi \cdot \frac{D_{tube}}{2} F_{t,s} \cdot n_t \cdot \Delta x \left(T_{s,x}^4 - T_{amb}^4\right) \quad (3.4)$$

Note that surface temperature is a function of position in the x-direction and cannot normally be evaluated in an integral without first defining its relationship with *x*. For this analysis, we have assumed that for each discrete element of length *Δx*, the surface temperature and other properties are constant. This allows the evaluation of the integrals with respect to *x* with temperatures and properties that are constant. The validity of this assumption is discussed in an upcoming section.

The convective losses $\dot{q}_{conv}$ are proportional to the temperature difference between the external surface of the receiver tube and the free stream air temperature, T_{amb}. Properties of air are evaluated at the film temperature, where the film temperature is the average of the tube surface and the ambient air temperature. The convective losses are proportional to a mixed convection coefficient h_m. This coefficient incorporates both natural and forced convection from the receiver surface, and is discussed further in the following section.

$$\begin{aligned} \dot{q}_{conv}(x) &= h_m \cdot D_{tube} \cdot n_t \int_{x_o}^{x_o+\Delta x} \left(T_s(x) - T_{amb}\right) dx \\ \dot{q}_{conv,x} &= h_m \cdot D_{tube} \cdot n_t \cdot \Delta x \cdot \left(T_{s,x} - T_{amb}\right) \end{aligned} \tag{3.5}$$

The final term is the thermal energy that is added to the heat transfer fluid, and is described by the $\dot{q}_{fluid}$ term. Since all other heat terms have been determined, the remaining unknown is the change in temperature of the heat transfer fluid across the control volume.

$$\begin{aligned} \int_{x_o}^{x_o+\Delta x} dT_{htf} &= \int_{x_o}^{x_o+\Delta x} \frac{\dot{q}_{fluid}(x)}{\dot{m}_{htf} \cdot c_{htf}(x)} dx \\ T_{htf,x_o+\Delta x} - T_{htf,x_o} &= \frac{\dot{q}_{fluid,x} \cdot \Delta x}{\dot{m}_{htf} \cdot c_{htf,x}} \end{aligned} \tag{3.6}$$

To determine the average surface temperature of a receiver tube, an additional energy balance is required. This balance considers the thermal resistance between the outer surface of the tube and the heat transfer fluid running through the tube. Figure 26

presents this balance imposed on a cross-section view of the receiver tube. The conduction resistance and convective resistance between the inner tube wall and the heat transfer fluid are shown, and can be calculated as described in Eqs.(3.7) and (3.8). The heat flux into the control volume ($\dot{q}_{in}$) represents the flux entering the tube after outer surface losses have been accounted for.

$$R_{cond} = \frac{\ln\left(\frac{D_{tube}/2}{D_{inner}/2}\right)}{\left(\pi \cdot 2 \cdot \Delta x \cdot k_{tube} \cdot n_t\right)} \tag{3.7}$$

$$R_{conv} = \frac{1}{h_{inner} \cdot \Delta x \cdot D_{inner} \cdot \pi/2 \cdot n_t} \tag{3.8}$$

The heat transfer to the fluid and the associated temperature rise of the fluid across the finite length can be calculated by using these thermal resistances in series. The driving temperature difference for the heat transfer is the difference between the surface temperature and the average heat transfer fluid temperature at position *x*. This relationship is shown in Eq. (3.9).

$$T_{s,x} - T_{htf,ave,x} = \dot{m}_{htf} \cdot c_{htf,x} \cdot \left(T_{htf,x_o+\Delta x} - T_{htf,x_o}\right)\left(R_{cond} + R_{conv}\right) \tag{3.9}$$

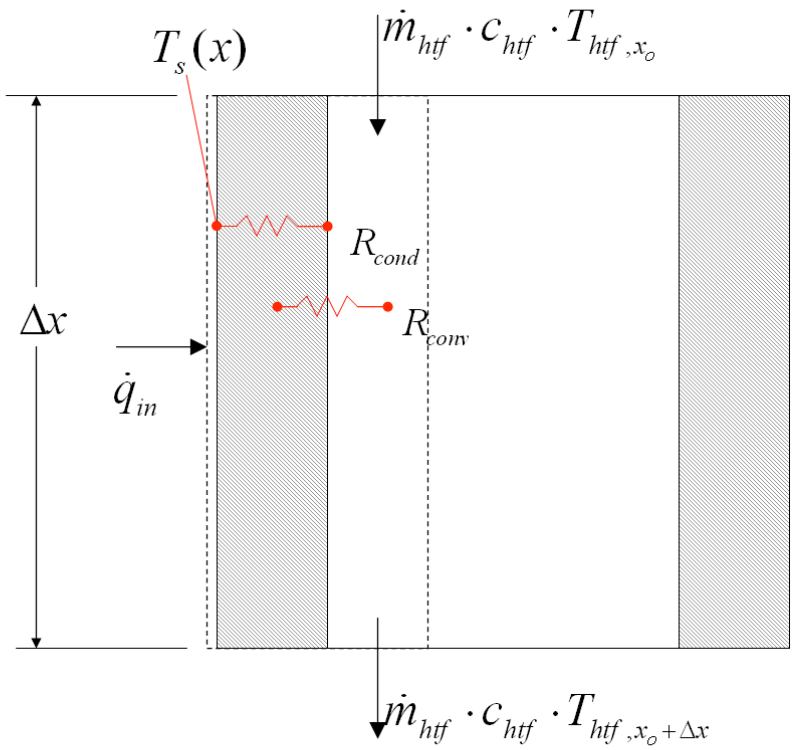


Figure 26: Heat flow balance across the receiver tube wall

Since several of the relationships described in the preceding equations are implicit in temperature, the analysis must begin with a set of guess values that can later be adjusted through iteration. The set of guess values includes the heat transfer fluid outlet temperature and tube surface temperature for each node. Additionally, the heat transfer fluid mass flow rate must be assigned a guess value since the mass flow rate is constrained to match the desired output temperature from the receiver. The affected implicit relationships include material properties like conductivity, density, and heat capacity, but also include driving temperature differences such as the surface temperature of the tube wall.

3.2.2 Evaluation of Loss Coefficients

The success of the technique provided in the previous section for constructing a thermal model of the central receiver depends in large part on the predictive accuracy of the thermal loss coefficients. This section discusses the process of formulating these

coefficients to account for temperature dependence, and in the case of convection, to account for competing loss mechanisms.

The three major mechanisms for thermal loss from the central receiver are reflection of the incident flux, radiation from the receiver surface to its surroundings, and convection (natural and forced) to the ambient. Reflection losses are estimated by specifying the absorptivity constant of the paint coating. Radiation losses are somewhat more complex, in that they are highly dependent on both the surface temperature of the receiver and the temperature(s) of the surroundings.

The radiation losses are the dominant heat transfer mechanism at high temperatures, so careful evaluation of their magnitudes is of particular importance. To determine the appropriate ambient temperature for use in the loss calculation, both the ambient air temperature and the effective sky temperature are considered, with appropriate view factors in place between the tower and the temperature zones. Equations(3.10)-(3.12) show these relationships.

$$\bar{h}_{rad,amb[i,j]} = \sigma\varepsilon F_{s,amb}\left(T_{s[i,j]}^2 + T_{amb}^2\right)\left(T_{s[i,j]} + T_{amb}\right) \tag{3.10}$$

$$\bar{h}_{rad,sky[i,j]} = \sigma\varepsilon F_{s,sky}\left(T_{s[i,j]}^2 + T_{sky}^2\right)\left(T_{s[i,j]} + T_{sky}\right) \tag{3.11}$$

$$\dot{q}_{rad[i,j]} = \bar{h}_{rad,amb[i,j]} \cdot A_s\left(T_{s[i,j]} - T_{amb}\right) + \bar{h}_{rad,sky} A_s\left(T_{s[i,j]} + T_{sky}\right) \tag{3.12}$$

The Stefan-Boltzmann constant is $\sigma = 5.67\times10^{-8}\left[\frac{W}{m^2K^4}\right]$ and the surface emissivity is $\varepsilon = .88$ as given in Taumoefolau, et al. (2004). The view factors between the ground and the tower, and the sky and the tower are $F_{s,amb} = 0.5$ *and* $F_{s,sky} = 0.5$, respectively. Finally, the sky temperature is estimated using the following relationship (Duffie and Beckman, 2006):

$$T_{sky} = T_{amb} \cdot \left(0.711 + 0.0056 \cdot T_{dewpt} + 0.000073 \cdot T_{dewpt}^2 + 0.013 \cdot \cos\left(\left(\frac{180 - hour \cdot 15}{180} \right) \pi \right) \right)^{\frac{1}{4}} \quad (3.13)$$

Convective losses from the power tower present an unusual heat transfer situation since the large diameter of the receiver results in large Reynolds number. At the very high Reynolds numbers present in this system, the traditional Nusselt correlations for heat transfer from a cylinder that are provided in heat transfer textbooks and in EES are not adequate since their formulation is applicable for lower Reynolds numbers typical of much smaller cylinder diameters. To accommodate this system, a correlation specific to this application by Siebers and Kraabel, (1984) is used.

Due to the abnormally large geometry of the external cylinder, natural convection from the receiver surface can be modeled using a vertical flat-plate correlation. The best relationship for this application appears to be that presented by Siebers and Kraabel, 1984 and given by:

$$Nusselt_{nat} = 0.098 \cdot Gr_H^{1/3} \left(\frac{T_s}{T_{amb}} \right)^{-0.14} \quad (3.14)$$

The Grashof number is evaluated where g is the gravitational constant, β is the volumetric expansion coefficient, ν_{amb} is the kinematic viscosity, and $T_{s,ave}$ is the average surface temperature of all of the elements in the panel. Fluid properties should be evaluated at the ambient temperature.

$$Gr_{nat} = g \cdot \beta \cdot \left(T_{s,ave} - T_{amb} \right) \cdot \frac{H_{rec}^3}{\nu_{amb}^2} \quad (3.15)$$

A comparison of this correlation and other correlations for similarly shaped geometry is presented in Figure 27. It should be noted that in the case of the Rohsenow (Nellis and Klein, 2009) flat-plate correlation, the Rayleigh number for this geometry is outside of

the applicable range. Fluid properties for these relationships should be evaluated at the film temperature.

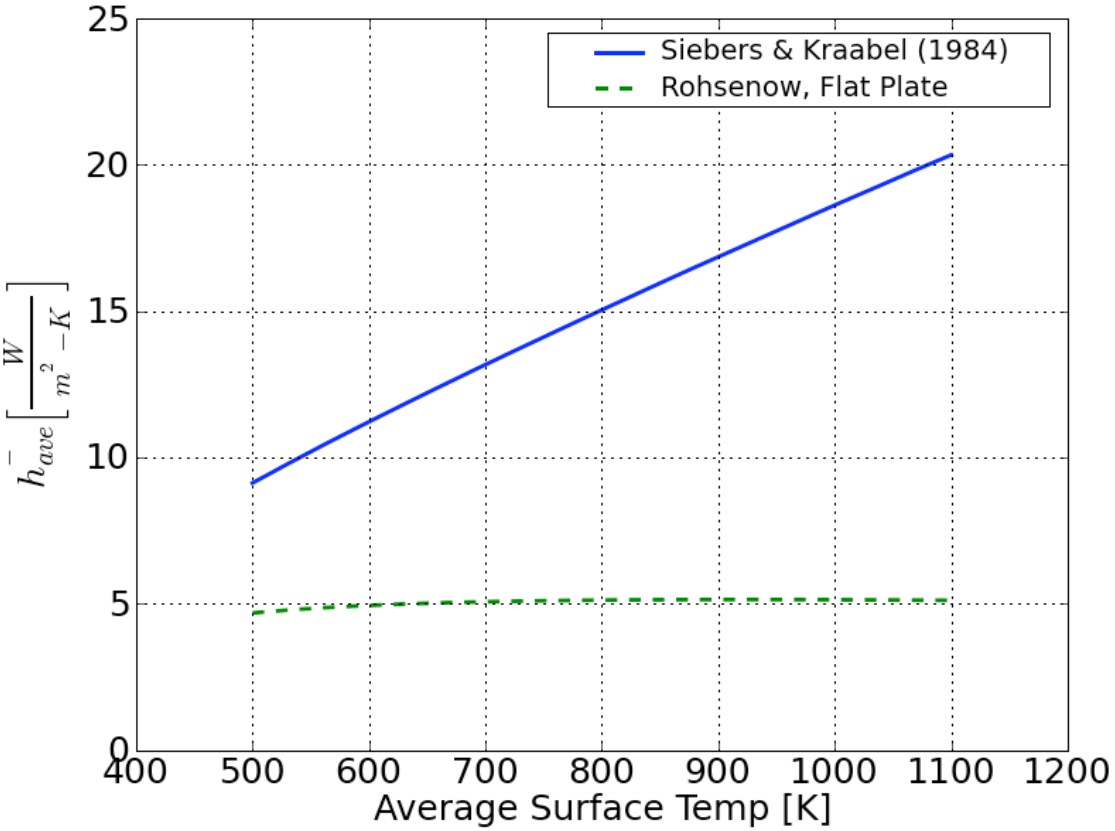


Figure 27: Natural convection as a function of surface temperature for the Siebers & Kraabel vertical cylinder correlation as compared a standard flat-plate correlation.

The forced convection correlation is provided as a set of curves that are applied for a specific range of Reynolds number and for a specific surface roughness. The correlation in all of its components is provided in Table 10 below.

Table 10: Nusselt number calculation for the external receiver in forced convection

	Reynolds Number Range	**Correlation**
$k_s / D = 0$ (A smooth cylinder)		
(1)	All Re	$Nu = 0.3 + 0.488 \cdot \mathrm{Re}^{0.5} \left(1 + \left(\frac{\mathrm{Re}}{282000} \right)^{0.625} \right)^{0.8}$
$k_s / D = 75 \times 10^{-5}$		
(2)	$\mathrm{Re} \le 7.0 \times 10^5$	Use smooth cylinder correlation entry (1)
(3)	$7.0 \times 10^5 < \mathrm{Re} < 2.2 \times 10^7$	$Nu = 2.57 \times 10^{-3} \cdot \mathrm{Re}^{0.98}$
(4)	$\mathrm{Re} \ge 2.2 \times 10^7$	$Nu = 0.0455 \cdot \mathrm{Re}^{0.81}$
$k_s / D = 300 \times 10^{-5}$		
(5)	$\mathrm{Re} \le 1.8 \times 10^5$	Use smooth cylinder correlation entry (1)
(6)	$1.8 \times 10^5 < \mathrm{Re} < 4.0 \times 10^6$	$Nu = 0.0135 \cdot \mathrm{Re}^{0.89}$
(7)	$\mathrm{Re} \ge 4.0 \times 10^6$	Use entry (4)
$k_s / D = 900 \times 10^{-5}$		
(8)	$\mathrm{Re} \le 1.0 \times 10^5$	Use smooth cylinder correlation entry (1)
(9)	$\mathrm{Re} > 1.0 \times 10^5$	Use entry (4)

The surface roughness can be estimated by taking the diameter of a single receiver tube over the diameter of the external receiver. If the surface roughness is calculated to be above the maximum value in the table of $k_s / D = 900 \times 10^{-5}$, the upper limit provided in entry number 4 can be used. Fluid properties should be evaluated at the film temperature.

Figure 28 shows the forced convection Nusselt number calculated with the Siebers & Kraabel correlation and with the Churchill and Bernstein (Nellis and Klein, 2009) correlation as a function of wind velocity. A surface roughness of 2.5×10^{-3} was used.

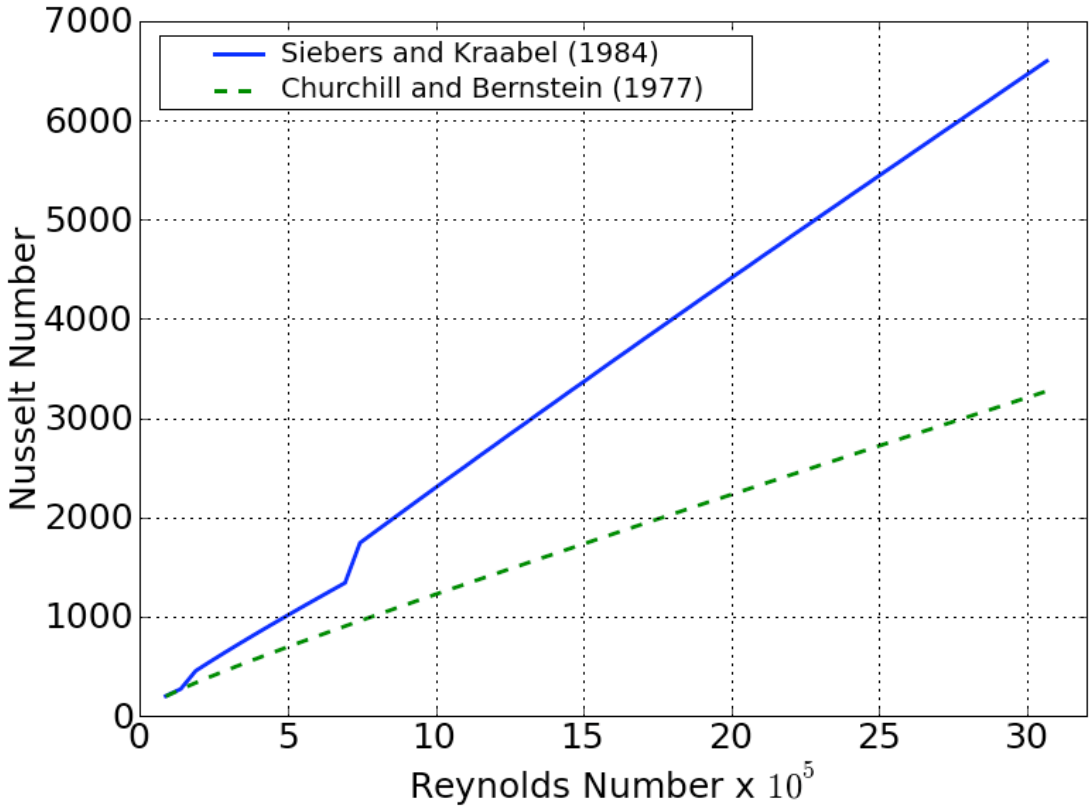


Figure 28: The forced convection coefficient as a function of Reynolds number as calculated with the Churchill and Bernstein correlation and with the Siebers and Kraabel correlation

As shown in Figure 12, the Siebers and Kraabel correlation results in a higher heat transfer coefficient at wind velocities above 2 m/s when compared to the Churchill and Bernstein correlation. Using the Siebers and Kraabel correlation in place of the original will ensure a more conservative model in terms of thermal efficiency of the receiver. One noteworthy feature of the Siebers and Kraabel correlation is the apparent discontinuities that occur. These are caused by the transition from one correlation to another at a particular value of the Reynolds number. This discontinuity does not reflect any physical behavior, but is essentially a side effect of splicing different correlations.

The combination of forced and natural convection is applied to the receiver using the mixed convection relationship. Mixed convection is generally understood to behave according to the relationship:

$$h_{mixed} = \left(h_{nat}^{m} + h_{for}^{m}\right)^{\frac{1}{m}} \tag{3.16}$$

The value of *m* indicates the degree of dominance of the larger convection coefficient over the smaller. As the value of *m* increases, the result will tend to be completely dominated by the larger of the two coefficients. According to the results of a detailed discussion in Siebers and Kraabel (1984), the value of *m* is selected to be *3.2*, indicating a relatively strong dependence on the larger of the two convection phenomenon. The mixed convection result appears as shown in Figure 29 for varying values of wind speed as compared with mixed convection using the Churchill and Bernstein equation for forced convection and the Rohsenow equation for natural convection from a flat plate.

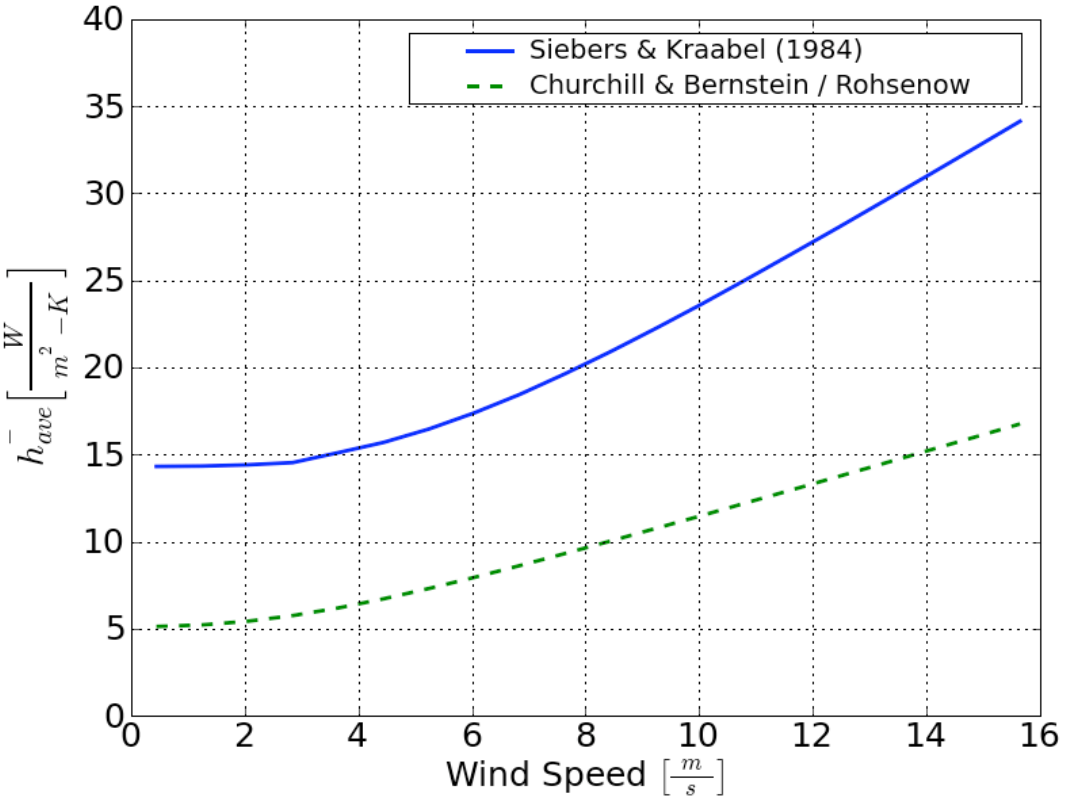


Figure 29: Mixed convection as a function of wind-speed using the Churchill and Bernstein / Rohsenow correlations and the Siebers and Kraabel correlations (m=3.2, D_{rec}=8.25[m])

3.2.3 Simplifications to the model

An initial model was developed using the relationships and correlations described in the previous section using EES. The purpose of this model was to both test the functionality of the model, and to determine areas in which the model could be improved for implementation in Fortran. This section provides an overview of the outcome of this work, and further constructs the foundation for the final Fortran model presented in upcoming discussion.

Computational speed is of interest in simulations that provide hourly evaluations over the course of one or more years. A system that must iterate to solve the governing equations at each time step is especially a concern. The receiver model employs many implicit equations, especially when the material and fluid properties are assumed to vary as a function of temperature. To address this issue, a sensitivity analysis was conducted using an EES receiver model to evaluate the effect of varying fluid and material properties over the extent of the receiver flow path.

Additionally, the study considered the effect of reducing the number of computational nodes of length Δx per receiver panel from ten to one. In other words, the effect of a two-dimensional flux distribution as compared to a one-dimensional distribution was considered. The one-dimensional flux data points were the average value of the flux distribution in the vertical direction on the receiver. Averaging these values resulted in one flux data point per panel on the receiver. A representative sample of the results from this study is presented in Table 11.

The "Air Constant" and "Coolant Constant" columns indicate whether the air and coolant properties were evaluated as a function of temperature for each node, or whether they were held constant at a provided temperature. The variable, $T_{htf,hot}$, represents the temperature of the salt leaving the system. The nominal temperature for the Solar II plant was 565 °C (838 K), so the flow rate of *75 kg/s* was chosen to bring the outlet temperature to approximately similar conditions. The $\eta_{thermal}$ column indicates the

thermal efficiency of the receiver considering all heat transfer losses. Finally, the *Comments* column provides the temperature at which the properties were evaluated for each case in which they were held constant.

Table 11: Sensitivity study examining the effect of model simplifications on output

Number of nodes	Coolant Constant?	Air Constant?	$T_{htf,hot}$	$\dot{Q}_{rad}$	$\dot{Q}_{conv}$	$\eta_{thermal}$	Solve Time	Comments
			K	MW	MW	-	sec	
10	Yes	Yes	856.4	3.13	0.642	0.946	8.1	Fluid at 700[K], Air at 550 [K]
10	Yes	No	856.4	3.13	0.637	0.946	8.1	Fluid at 700[K]
10	No	Yes	856.3	3.13	0.645	0.946	74.3	Air at 550[K]
10	No	No	856.4	3.13	0.641	0.946	45.9	
1	Yes	Yes	856.5	3.11	0.641	0.946	1.0	Fluid at 700[K], Air at 550 [K]
1	Yes	No	856.5	3.11	0.637	0.946	0.8	Fluid at 700[K]
1	No	Yes	856.4	3.11	0.645	0.946	6.1	Air at 550[K]
1	No	No	856.5	3.11	0.641	0.946	2.6	

It is clear from the data in Table 11 that the model can be safely simplified to one node per panel at constant fluid properties. It should be noted that the resulting temperature and efficiency are sensitive to the temperature used for evaluating property data. A carefully considered evaluation temperature will lead to a successful simplification of the analysis. It is recommended that the evaluation temperature for the coolant fluid be the average of the heat transfer fluid temperatures at the receiver inlet and outlet.

The specification of both of these temperatures as input information will make this choice a valid simplification. Because the surface temperature of the receiver is not specified at runtime, the film temperature can be specified using the outlet salt temperature and the ambient temperature, both of which are known.

With these simplifications, the model can be implemented in Fortran and eventually as a TRNSYS component. The adverse effects of reducing the number of nodes from 10 to 1 per panel are most evident in the loss of the ability to calculate the surface temperature distribution along the vertical axis of the receiver.

3.3 Fortran Receiver Model

This section provides a detailed layout and discussion of the Fortran receiver model. This model makes use of the theoretical foundation provided in the *3.2.1 Energy Balance and Model* Formulation section, and also makes use of the correlations and simplifications discussed above. The following discussion is structured to provide illustrative narrative and background while directly presenting the Fortran receiver model code (shaded lines). A complete listing of the program code can be found in the electronic supplement accompanying this paper.

The model is structured as a set of Fortran subroutines that can be called by TRNSYS or by a special-purpose program. The main subroutine file includes the bulk of the thermal model, while the remaining subroutines are supplemental and perform specific functions when called by the model. Whether the calling program is TRNSYS or another code, the model requires that a certain set of inputs and parameters is provided, while returning a fixed set of outputs. These inputs, parameters, and outputs are listed in Table 12.

Table 12: The inputs, parameters, and outputs of the Fortran central tower receiver model

Input	Description	Default value	Units	Min/Max
azimuth	The solar azimuth angle (deg). By convention, due South is 0 degrees. To the west of due south is (+) to 180 at North and to the east is (-) down to -180 at North.	0	Degrees	[-180,180)
zenith	The solar zenith angle (deg) defined as the angle between the solar vector and the vector normal to the surface of the earth.	10	Degrees	[0,100)
Outlet fluid temp	The temperature set point of the outlet salt. Mass flow demand of the coolant will be calculated accordingly (K)	838	K	(200,2500)
Inlet fluid temp	The temperature of the salt at the inlet of the receiver (K)	565	K	(250,1000)
Wind velocity	The receiver-level wind velocity	4.87	m/s	[0,100]
Ambient pressure	The ambient pressure condition at the receiver	1	atm	[0.1,2.0]
Pump efficiency	The pump isentropic efficiency times the mechanical conversion efficiency of the motor	0.65	-	[0,1]
Hour of the day	The hour of the day for the timestamp, *12* represents the value of solar noon	0	hr	(0,24]
Dew point	The dew point temperature at the location of	0	°C	[-75,50]

temperature	interest			
Direct Normal radiation	The incident solar flux on the heliostat field. Equal to Hourly Direct Normal Insolation/3.6	0	(W/m^2)	[0, 1500]
Field eff	The net field efficiency as a function of solar azimuth and zenith angle. Note that this is the total optical and reflective efficiency of the heliostat field, and does not include weather or receiver effects.	0	-	[0,1]
Dry Bulb Temperature	The dry-bulb temperature at the location of interest	15	°C	[-50,50]
Parameter	**Description**	**Default value**	**Units**	**Min/Max**
Number of Panels	The number of vertical panels constituting the receiver	16	-	(0,1000]
Receiver diameter	The total diameter of the receiver	8.15	m	(0,50]
Panel Height	The height of the receiver panels. This equals the height of the receiver	6.2	m	(0,50]
Tower height	The height of the tower structure above the plane of the ground.	120	m	[10,500]
Tube outer diameter	The outer diameter of the individual receiver panel tubes	25	mm	(0,1000]
Tube thickness	The thickness of the individual tube walls	1.24	mm	(0,100]
Material	An integer indicating the receiver tube material. The material can be any of the following: [1] – 'Stainless_AISI316'	1	-	(0,1]
Coolant	An integer that indicates the type of heat transfer fluid passing through the receiver. The coolant can be any of the following: [1] - 'Salt (68% KCl, 32% MgCl2)' [2] - 'Salt (8% NaF, 92% NaBF4)' [3] - 'Salt (25% KF, 75% KBF4)' [4] - 'Salt (31% RbF, 69% RbBF4)' [5] - 'Salt (46.5% LiF, 11.5%NaF, 42%KF)' [6] - 'Salt (49% LiF, 29% NaF, 29% ZrF4)' [7] - 'Salt (58% KF, 42% ZrF4)' [8] - 'Salt (58% LiCl, 42% RbCl)' [9] - 'Salt (58% NaCl, 42% MgCl2)' [10] - 'Salt (59.5% LiCl, 40.5% KCl)' [11] - 'Salt (59.5% NaF, 40.5% ZrF4)' [12] - 'Salt (60% NaNO3, 40% KNO3)'	1	-	(0,12]
Flow Pattern	An integer that indicates the mode of flow pattern to be used. A description of each flow type appears below: [1] This flow pattern begins at the northmost 2 panels, splits into 2 flows, and crosses over at the quarter position, exiting in 2 flows on the southmost 2 panels. This is the flow configuration that was	1	-	(0,8]

	used for SOLAR II [2] This flow pattern is the same as flow pattern #1, but in reverse. The salt enters on the 2 southmost panels, crosses over, and exits on the 2 northmost panels. [3] This flow pattern has 2 separate flows that enter in 2 of the northmost panels and flow around (without crossing over), exiting at the 2 southmost panels [4] This flow pattern has 2 separate flows that enter in 2 of the southmost panels and flow around (without crossing over), exiting at the 2 northmost panels [5] This flow type enters on a panel at the southmost side of the receiver, travels completely around the receiver in a clockwise direction, and exits again on the south side [6] This flow type enters on a panel at the southmost side of the receiver, travels completely around the receiver in a counter-clockwise direction, and exits again on the south side [7] This flow type enters on a panel at the northmost side of the receiver, travels completely around the receiver in a clockwise direction, and exits again on the north side [8] This flow type enters on a panel at the northmost side of the receiver, travels completely around the receiver in a counter-clockwise direction, and exits again on the north side			
Plant latitude	The latitude of the power plant. This can also be obtained from the TMY2 file.	35.0	Degrees	[-90,90]
Logical Unit	The logical unit of the 'fluxmap.csv' file. This will be automatically set by TRNSYS at runtime.	19	-	(0,+∞)
Output	**Description**	**Default value**	**Units**	**Min/Max**
Salt flow rate	The total mass flow rate of the heat transfer fluid through the receiver	150	kg/s	[0,2000]
Receiver thermal eff	The thermal efficiency of the receiver defined as: $1-\frac{radiation+convection}{incident\ power}$ Does not include reflection/optical losses from the field, since these are included in the heliostat efficiency value provided to the receiver.	0.95	-	[0,1]
Pump power	The total electric power required to provide a flow of $\dot{m}_{tot}$ through the receiver.	0	W	[0,+∞)
Convection losses	The total instantaneous convective losses from the receiver surface to ambient air.	$1.0x10^6$	W	[0,+∞)
Radiation losses	The total instantaneous radiation losses from the surface of the receiver to the ambient.	$4.0x10^6$	W	[0,+∞)
Thermal power	The total thermal power at the outlet of the	0	W	(-∞,+∞)

	receiver. This power is based on the total mass flow rate, the specific heat of the coolant, and the temperature difference between the outlet and the storage temperature of the salt. $\dot{q}_{thermal} = \left(T_{htf,hot} - T_{htf,cold}\right)\dot{m}_{htf} c_{htf}$			
HTF Outlet Temp	The outlet temperature of the heat transfer fluid. This output echoes the specified input temperature.	838	K	[200,1500]

The model begins with various variable declarations, however these have been omitted to improve clarity. The first several lines deal with converting the input variables from their original format to base SI units. Simple calculations are done to determine geometrical dimensions that have not been specified.

```
!Receiver dimensions, parameters
D_tube = D_out*myconvert('mm','m')      !Value in meters
th_tube = th_tu*myconvert('mm','m')     !Thickness of the tube
D_inner = D_tube - 2.*th_tube           !Diameter of each receiver tube
D_in = D_inner*myconvert('m','mm')      !Inner diameter of the tube
A_tube = pi*D_tube/2.*H_rec             !Outer area of each tube
n_t = Nint((pi*D_rec)/(D_tube*N_panels))!The number of tubes per panel
N_tube =n_t*N_panels                    !Number of tubes in the system
A_receiver = pi**2*D_rec*H_rec          !The area of the receiver exposed
                                         to the ambient air, [S&K]
A_rec_proj = D_tube*H_rec*N_tube        !The projected area of the tubes
T_amb = converttemp('C','K',T_atm)      !Ambient temperature
T_sky = skytemp(T_amb,(T_dp+273.15),hour)!The effective sky temp [K]
A_node = pi*D_rec/N_panels*H_rec        !The area of each node
P_atm=P_amb*myconvert('atm','Pa')       !Ambient pressure, in [Pa]
azi_adj = azimuth + 180.                !By TRNSYS convention, the
azimuth angle is 0 at due south, negative to the east, and positive to
the west. The range is then -180 to 180. By the convention used here,
the azimuth is 0 at due north, and ranges clockwise from 0 to 360. This
adjusts.
!Other parameters which are useful for calculating heat flow
sigma = 5.670e-8          ![W/m^2-K^4] Stefan-Boltzmann constant
epsilon = .88             !Emissivity: Taumoefolau, T. et al., 2004
```

Several functions were written specifically for uses in this code, including a conversion script *myconvert(units1,units2)* that returns a multiplying factor to convert the value from the first unit type to the second. A similar function is used to convert temperatures, called *converttemp(units1,units2)*. The final user-defined function used is the *skytemp* function. This provides the effective sky temperature for a given dew point temperature, ambient temperature, and atmospheric pressure based on the relationship developed in Duffie and Beckman (2006).

To avoid needless iteration during night times of little or no solar availability, a check is performed to queue a subroutine exit if minimum availability criteria are not met. A compound *if* statement is used for this test, returning the subroutine output values as zero to the calling program. A practical limitation for the solar zenith angle was found to be 85°, or just above the horizon, while the direct normal irradiation (DNI) limit was set to be at or above $150 W/m^2$. These provisions were found to greatly increase the likelihood of model convergence, and eliminate non-physical results.

```
!Do an initial check to make sure the solar position called is valid.
! If its not, return the output equal to zeros. Also check to make sure
! the solar flux is at a certain level, otherwise the model isn't valid
if(((zenith.gt.85.).or.(I_bn.lt.150.)).or.&
   ((zenith.eq.0.).and.(azimuth.eq.0))) then
    m_dot_salt_tot=0 ; eta_therm=0 ; W_dot_pump=0
    q_conv_sum=0 ; q_rad_sum=0 ; T_s=0 ;  Q_thermal=0
    return
endif
```

As was discussed in a previous section, the receiver panels can be arranged in any combination of series and parallel flow paths. Eight common and geometrically symmetric flow paths have been programmed into this model, and are defined within the *flowPatterns* subroutine. A call to this code requires inputs including the number of receiver panels (*N_panels*), a number corresponding to the desired flow-type (*flowtype*), and returns outputs with an array indicating the flow position of each panel (*Flow_pattern*), the panels where the heat transfer fluid leaves the receiver (*salt_out*), and the number of independent flow paths (*nlines*).

```
!Get flow pattern
call flowPatterns(N_panels,flowtype,Flow_pattern,salt_out,nlines)
```

Another subroutine is defined to define the heliostat field flux distribution on the receiver. The *fluxinterp* subroutine takes the logical unit of the flux-map file (*LU_flux*) and the solar zenith (*zenith*) and azimuth (*azi_adj*) angles, and searches the file to find the most closely matching flux distribution. This is returned to the program as an $[12 \times 1]$ array (*array*) along with the associated day of the year (*daycall*) and hour of the day

(*hourcall*) for reference. The code for these and other central receiver component functions and subroutines are listed in the electronic supplement accompanying this paper.

```
!Get the values of the flux from the fluxmap
call fluxinterp(LU_flux,zenith,azi_adj,array,daycall,hourcall)
```

These flux values are provided to the receiver model via the PTGen program, which in turn makes use of the DELSOL3 code to calculate flux distribution as described in Chapter 5. DELSOL3 has been constructed to provide flux values without regard to net field efficiency, and at an assumed DNI of $950 W/m^2$. To make use of the flux distribution, the data points are normalized according to the nominal flux. The actual DNI and field efficiency can then be used to determine the flux distribution on the receiver.

```
do j=1,12
  flux_in(j)=array(j)/(950.)*I_bn*field_eff  !The weather-adjusted,
efficiency-adjusted flux values
enddo
```

Although the flux array contains 12×1 nodes, the number of panels on the receiver can by any number. The most practical way to implement the flux distribution in the model is to provide a flux value for each receiver panel. This is accomplished by imposing the range of panels on a scale from one to twelve, and then interpolating between data points when needed. Figure 30 illustrates this technique for 16 panels.

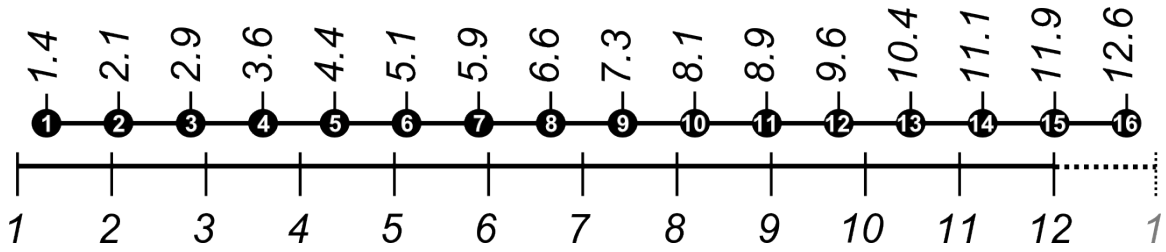


Figure 30: A technique for imposing a flux distribution on a different number of receiver panels. Sixteen panels are distributed among twelve available flux data points.

Since the geometry of the receiver is a cylinder, node 12 and node 1 are actually adjacent. Therefore, the imposed numbering system must be able to interpolate between these two nodes when the flux distribution is applied to a different number of panels. This interpolation is shown in Figure 30. The imposed distribution must also be centered around the center of the data point distribution, or the flux will be skewed to falsely suggest that one side of the receiver receives more flux over the course of the simulation. The code for this technique is shown below.

```
!Translate to the number of panels, so each panel has its own linearly
interpolated flux value
do j=1,N_panels
  Panel(j) = j                              !The position of each panel
  ppos(j)=(12./real(N_panels)*(j-1.)+6./real(N_panels))+1.
  flo(j)=floor(ppos(j))
  ceil(j)=ceiling(ppos(j))
  ind(j)= (ppos(j)-flo(j))/(ceil(j)-flo(j))
  if(ceil(j).gt.12) then
    ceil(j)=1
  endif
  Psp_field(j)=ind(j)*(flux_in(ceil(j))-
flux_in(flo(j)))+flux_in(flo(j))   !Average area-specific power for node
  P_field(j)=A_node*Psp_field(j)   !The power incident on each node
enddo
```

With the flux distribution, flow configuration, and input values established, the iterative portion of the thermal model is undertaken. This section includes calculations to determine the receiver performance, and is solved by using successive substitution. To keep the calculated values and the guess values for each run separate, the guess values for each run are denoted with suffix "*X*".

```
!Guess values ------------------------------------------------------
do j=1,N_panels
    T_sX(j) = 800.              !Guess temperature for the surface nodes
    T_panel_outX(j) = 600.      !Guess values for the fluid temp coming
                                 out of the control volume
    T_panel_inX(j) = 600.       !Guess values for the fluid temperature
                                 coming into the control volume
enddo
m_dot_saltX = 180./real(nlines)    !coolant mass flow rate guess value
T_salt_hotX = 9999.                !Initial value for error calculation
!-------------------------------------------------------------------
```

These guess values will be updated through the course of the calculations. Convergence in the model is determined by checking the relative difference between the actual hot heat transfer fluid outlet temperature and the set-point outlet temperature provided by the user. If convergence is not obtained in a specified number of tries, then the problem is unlikely to ever converge with the given inputs, and the subroutine returns with zero values for the outputs.

```
do qq=1,201
  !if the problem fails to converge after 200 iterations, then the
power is likely negligible and the zero set can be returned
  if(qq.gt.200)then
    m_dot_salt_tot=0 ; eta_therm=0 ; W_dot_pump=0
    q_conv_sum=0 ; q_rad_sum=0 ; T_s=0 ;  Q_thermal=0
    return
  endif
  err=(T_salt_hotX - T_salt_hot)/T_salt_hot  !Convergence error
  if(abs(err).lt.(1.0e-4)) exit              !Check for convergence
  !Set the variables equal to their newly calculated guess values------
  do j=1,N_panels
    T_s(j) = T_sX(j)
    T_panel_out(j) = T_panel_outX(j)
    T_panel_in(j) = T_panel_inX(j)
  enddo
  m_dot_salt = m_dot_saltX
  !---------------------------------------------------------------------
```

Using the guess values for either the first pass or any following iteration, several temperatures needed for further calculation are determined. These include the average panel heat transfer fluid temperature (*T_panel_ave*), the air-receiver film temperature (*T_film*), and an overall average heat transfer fluid temperature. Also needed are overall average temperatures (*T_s_ave, T_film_ave*), for use in the forced convection correlation.

```
!Now do the actual calculations
do j=1,N_panels
  T_panel_ave(j) = (T_panel_in(j)+T_panel_out(j))/2.  !The average
coolant temperature in each control volume
  T_film(j) = (T_s(j)+T_amb)/2.
enddo
T_coolant_prop = (T_salt_hot + T_salt_cold)/2.        !The temperature
at which the coolant properties are evaluated. Validated as constant
T_salt_hotX = sum(T_panel_out(salt_out(1:nlines)))/real(nlines)
!Calculates the mixed outlet temperature of the salt
!Calculate the average surface and film temperature so that the forced
convection coefficient can be determined
T_s_ave = sum(T_s(1:N_panels))/(real(N_panels))
```

```
T_film_ave = (T_amb+T_salt_hot)/2.
```

The forced convection losses from the receiver are applied based on the total geometry of the receiver. Since no detailed information on local forced heat transfer coefficients is available, a total average coefficient will be used. Determination of the total average coefficient requires the use of the total average surface temperature and total average film temperature calculated in the preceding code. The surface roughness used for the receiver (k_s / D) can be approximated using the radius of a single receiver tube over the total external diameter of the receiver. Finally, the forced convection heat transfer coefficient is calculated according to the relationship previously defined in Table 10.

```
!Convection coefficient for external forced convection using S&K
k_film = Conductivity('Air',T_film_ave)      !Conductivity of the air
mu_film = Viscosity('Air',T_film_ave)        !Air Dynamic viscosity
rho_film = Density('Air', T_film_ave, P_atm) !Density of the air
c_p_film = specheat('Air',T_film_ave)*myconvert('kJ','J')  !Spec.heat
Re_for = rho_film*V_wind*D_rec/mu_film       !Reynolds number
ksD = (D_tube/2.)/D_rec      !The effective roughness of the cylinder
Nusselt_for=Nusselt_FC(ksD,Re_for)           !Nusselt #
h_for = Nusselt_for*k_film/D_rec    !Heat transfer coefficient
```

The natural convection coefficient is calculated using air properties at ambient conditions using the Nusselt number correlation is defined above in Eqs.(3.14) and (3.15). This value can be calculated for each individual panel, modeled as a vertical flat surface.

```
!Convection coefficient for external natural convection
!Note: This relationship applies when the surrounding properties are
evaluated at ambient conditions
beta = volexpcoef('Air',T_amb)        !Volumetric expansion coefficient
nu_amb = Viscosity('Air',T_amb)/Density('Air',T_amb,P_atm) !Kinimatic
viscosity
do j=1,N_panels
  !Grashof #
  Gr_nat(j) = grav*beta*(T_s(j)-T_amb)*H_rec**3/nu_amb**2
  !Nusselt number
  Nusselt_nat(j) = .098*Gr_nat(j)**(1./3.)*(T_s(j)/T_amb)**(-.14)
  h_nat(j) = Nusselt_nat(j)*k_film/H_rec  !Natural convection coef.
enddo
```

Mixed convection is then applied using the mixed convection equation, with an exponent *m* equal to the selected value of 3.2 (Siebers and Kraabel, 1984). A coefficient is

calculated for each individual panel, using the universal forced convection coefficient and the panel-specific value of the natural convection coefficient. The convective heat transfer (*q_dot_conv*) is also calculated for each panel, using the projected area each panel multiplied by the surface to film temperature difference.

```
!Mixed convection
m=3.2
do j=1,N_panels
  h_mixed(j) = (h_for**m+h_nat(j)**m)**(1./m)
  q_dot_conv(j) = h_mixed(j)*A_node*(T_s(j) - T_film(j))   !Convection
enddo
```

Radiation from the receiver is also determined panel by panel. The radiation coefficient and heat transfer are broken into a portions reflecting the proportion of radiation exchange taking place with the surrounding ground and with the sky, expressed as an average effective sky temperature.

```
!Radiation from the receiver
    !Calculate the radiation node by node
do j=1,N_panels
  h_rad_amb(j) = sigma*epsilon*(T_s(j)**2+T_amb**2)*(T_s(j)+T_amb)
!The radiation coefficient for amb
  h_rad_sky(j) = sigma*epsilon*(T_s(j)**2+T_sky**2)*(T_s(j)+T_sky)
!The radiation coef. for sky
  q_dot_amb(j) = .5*h_rad_amb(j)*A_node*(T_s(j)-T_amb)  !amb losses per
node
  q_dot_sky(j) = .5*h_rad_sky(j)*A_node*(T_s(j) - T_sky)  !sky losses
per node
enddo
```

Total panel losses can now be determined, and these individual losses are summed for the entire receiver.

```
!Calculate the losses from the surface"
do j=1,N_panels
  q_dot_rad(j) = q_dot_amb(j)+q_dot_sky(j)   !Total rad.losses per node
  q_dot_loss(j) = q_dot_rad(j)+q_dot_conv(j) !Total losses per node
enddo
q_loss_sum = sum(q_dot_loss(1:N_panels))     !Receiver total losses
q_conv_sum = sum(q_dot_conv(1:N_panels))     !Receiver convection
q_rad_sum = sum(q_dot_rad(1:N_panels))       !Receiver radiation losses
```

Using the guess values provided at the initial stages of this model, the receiver losses have now been calculated. The incident radiation is determined, and the absorbed thermal power is then the incident radiation minus the losses. This completes the calculations for the external side of the receiver.

```
!Calculate the flux incident on the surface
do j=1,N_panels
  q_dot_inc(j)=P_field(j)*myconvert('kW','W')
  q_dot_abs(j) = q_dot_inc(j) - q_dot_loss(j)   !The absorbed flux
enddo
q_inc_sum = sum(q_dot_inc(1:N_panels))  !The total power incident
q_abs_sum = sum(q_dot_abs(1:N_panels))  !The total power absorbed
```

As heat flows from the external surfaces of the receiver toward the internal fluid, it passes through the thin tube wall. This thermal resistance can be calculated using the thermal conductivity of the material and the tube thickness. The thermal conductivity is a function of wall temperature, and this is reflected with the use of the property function *Conductivity* that was written as a part of this research.

Property functions have been defined using correlations for several potential heat transfer fluids and one receiver material as presented in Table 12 above. Due to the high velocity of the heat transfer fluid past the receiver tubes up to5 m/s, it is assumed that the wall temperature (*T_wall*) can be approximated as the average of the fluid temperature and the external surface temperature. It is important to note that this wall temperature is only used for the calculation of the tube conductivity material property, and the actual convective resistance between the fluid and the tube wall is considered in subsequent calculations.

```
!Calculate the temperature drop across the receiver tube wall.. assume
a cylindrical thermal resistance
do j=1,N_panels
  !The temperature at which the conductivity of the wall is evaluated
  T_wall(j) = (T_s(j) + T_panel_ave(j))/2.
  !The conductivity of the wall
  k_tube(j) = Conductivity(Material,T_wall(j))
  !The thermal resistance of the wall
  R_tube_wall(j) = th_tube/(k_tube(j)*H_rec*D_rec*pi**2/2.)
enddo
```

The convective heat transfer coefficient for the inner surface of the receiver tubes is defined using the Gnielinski equation for fully developed convection in a circular tube (Gnielinski, V., 1976). This is implemented in the *PipeFlow* subroutine written as a part of this research. In low-flow situations where the heat transfer fluid velocity does not provide enough convective heat transfer to the fluid, the Nusselt number correlation may return non-physical results. A safeguard is put in place, where an invalid Nusselt number will cause the program to exit assuming with the zero set. In calculating the convective resistance (*R_conv_inner*), a surface area of ½ the inner surface area is used since the back side of the tube is insulated and assumed adiabatic, and circumferential heat transfer by conduction along the tube wall is minimal.

```
!Calculations for the inside of the tube
C_p_coolant = specheat(Coolant, T_coolant_prop)*myconvert('kJ','J')
!Specific heat of the coolant
LoverD = H_rec/D_inner
RelRough = (1.5e-6)/D_inner      !Relative roughness of the tubes.
mu_coolant = viscosity(Coolant, T_coolant_prop)    !Absolute viscosity
k_coolant = conductivity(Coolant,  T_coolant_prop) !Conductivity
rho_coolant = density(Coolant, T_coolant_prop, 0.) !Density

!Average velocity of the coolant through the receiver tubes.
u_coolant = m_dot_salt/(n_t*rho_coolant*(D_inner/2.)**2*pi)
!Reynolds number for internal flow
Re_inner = rho_coolant*u_coolant*D_inner/mu_coolant
!Prandtl number for internal flow
Pr_inner = C_p_coolant*mu_coolant/k_coolant
!The internal convection correlation. Petukhov, Gneilinski
call PipeFlow(Re_inner,Pr_inner,LoverD,relRough,Nusselt_t,f)
if(Nusselt_t.le.0) then
    m_dot_salt_tot=0 ; eta_therm=0 ; W_dot_pump=0
    q_conv_sum=0 ; q_rad_sum=0 ; T_s=0 ;  Q_thermal=0
  return
endif
!Convection coefficient between the inner tube wall and the coolant
h_inner = Nusselt_t*k_coolant/D_inner
!The thermal resistance associated with this value
R_conv_inner = 1./(h_inner*pi*D_inner/2.*H_rec*n_t)
```

With heat addition and thermal resistances known for each panel, the heat transfer fluid temperatures can be calculated. Inlet temperatures for each panel are defined by the *Flow_Pattern* array discussed previously. This array provides an index indicating the flow source for each panel on the receiver. For example, in the case that panels *3* and *4* are adjacent, and the heat transfer fluid flows from *3* to *4*, *Flow_Pattern(4)* is returned as

3. In this way, the temperatures at the inlet of each panel can be specified. Panels that are the first to receive the cold heat transfer fluid at the receiver inlet are designated with a *Flow_pattern* index of *0*.

The updated guess outlet temperature for each panel (*T_panel_outX*) is calculated, and the average of this outlet temperature and the inlet temperature are used to establish the updated guess external surface temperature (*T_sX*). Recall that the *X* suffix denotes the guess value for the variable, which is substituted in for the actual value at the beginning of the iteration. The thermal efficiency of the receiver is the absorbed thermal power over the incident thermal power.

```
!Set up numerical flow grid
do j=1,N_panels
  !The panel inlet temp is equal to the panel outlet temp from the
  previous panel, according to the flow diagram
  if(Flow_pattern(j).lt.1) then
    T_panel_in(j) = T_salt_cold
  else
    T_panel_in(j) = T_panel_out(Flow_pattern(j))
  endif
  !The energy balance for each node
  T_panel_outX(j) = T_panel_in(j) + q_dot_abs(j)/
                    (m_dot_salt*c_p_coolant)
  !Calculate the surface temperature based on the absorbed heat
  T_sX(j)=T_panel_ave(j)+q_dot_abs(j)*(R_conv_inner+R_tube_wall(j))
enddo
eta_therm = q_abs_sum/q_inc_sum
```

The last step of the iterative block is the calculation of a new heat transfer fluid mass flow rate guess value. This is accomplished using an energy balance with the entire receiver as a control volume. Recall that the heat transfer fluid mass flow rate guess value (*m_dot_saltX*) is the flow rate through an individual receiver flow path. To reflect this, the balance is divided by the number of flow paths (*nlines*). If the mass flow rate is calculated to be a value low enough where the receiver model is unlikely to converge (in this case 5*kg/s*) the iteration is stopped and the subroutine returns with the zero set.

```
  !Final calculations
  m_dot_saltX = q_abs_sum/(nlines*C_p_coolant*(T_salt_hot-T_salt_cold))
  !Do a check to make sure the mass flow rate is reasonable
  if(m_dot_saltX.lt.5.) then
```

```
        m_dot_salt_tot=0 ; eta_therm=0 ; W_dot_pump=0
        q_conv_sum=0 ; q_rad_sum=0 ; T_s=0 ;  Q_thermal=0
        return
    endif
enddo
```

The iteration of the previous code continues until the error between the desired heat transfer temperature and the actual outlet heat transfer temperature is acceptable. At this point, several follow-up calculations can be done to obtain a better representation of the central receiver.

The total mass flow rate is determined by multiplying the individual flow path mass flow rate by the number of flow paths.

```
m_dot_salt_tot = m_dot_salt*nlines
```

The magnitude of the pressure drop associated with pumping the heat transfer fluid through up to the receiver and through it is required to obtain the associated pump parasitic power. The complex flow arrangement has been modeled as a series of pressure drops for straight and bent tubes using equivalent straight-tube lengths (Fox, et. al, 2006). The friction factor (*f*) corresponding the Nusselt number correlation for tubular flow is used. The total pressure drop is the sum of these individual components, and the pump power is scaled by a pump efficiency provided by the user. The thermal power associated with the receiver is also determined.

```
L_e_45 = 16. !The equivalent length produced by the bends in the tubes.
L_e_90 = 30.
!Pressure drop across the tube, straight length
DELTAP_tube = rho_coolant*(f*H_rec/D_inner*u_coolant**2/2.)
!Pressure drop across 45 degree bends
DELTAP_45 = rho_coolant*(f*L_e_45*u_coolant**2/2.)
!Pressure drop across 90 degree bends
DELTAP_90 = rho_coolant*(f*L_e_90*u_coolant**2/2.)
!Total pressure drop across the tube,(4)90-deg bends, (2)45-deg bends
DELTAP = DELTAP_tube + 2*DELTAP_45 + 4*DELTAP_90
!The pressure drop from pumping up to the receiver
DELTAP_THT = rho_coolant*THT*grav
!The net pressure drop across the receiver panels
DELTAP_net = DELTAP*N_panels/real(nlines)+DELTAP_THT
Pres_D = DELTAP_net*myconvert('Pa','MPa')
!The energy required by the pump to move coolant through the receiver
```

```
W_dot_pump = DELTAP_net*u_coolant*pi*D_inner**2/4.*n_t/eta_pump
!Calculate the thermal output of the tower
Q_thermal = m_dot_salt_tot*C_p_coolant*(T_salt_hot - T_salt_cold)
end subroutine
```

4 Rankine Cycle for the Central Receiver Power Plant

With the exception of a few niche technologies such as hydrogen generation and thermal process heat generation, the thermal power from the central receiver system is used to drive an electric power generation cycle. The most common generation cycle for this application is the steam Rankine cycle. As with any steam Rankine power generation cycle, the central receiver power cycle can include any variety of configurations to ease implementation and boost efficiency. The following discussion introduces a Rankine cycle modeling tool that is intended for general use with multiple cycle design configurations.

Solar energy-driven power cycles present a set of unique challenges and constraints. Unlike conventional base-load power cycles typical of a fossil-fuel fired plant that often see very little if any variation in operating conditions over the course of their operation lifetime, the CRS power cycle is subject to frequent transients in weather, daily shutdown and startup, and varying heat transfer fluid flow rates and temperatures. The ability of a long-term simulation to accommodate these variances is paramount to the accuracy of that simulation, since deviations from the design conditions of a power cycle adversely affect performance.

The task of producing a model or set of representative Rankine cycle models becomes especially daunting with the knowledge that plant to plant CRS power cycles may include an assortment of equipment designs, perhaps including multiple open or closed feed-water heaters, multiple turbine stages, regeneration, and pre-heat and super-heat of the steam. Nonetheless, this remains the goal of the research presented in the following discussion.

Several steps are taken to construct this examination, beginning with the establishment of a thermodynamic model in EES to define the sizing parameters of a representative power cycle. This model has been dubbed the “Design cycle,” since the outcome is a set of heat

exchanger conductances and fluid flow rates that are sized to match a desired electric power value. Next, a model is developed to evaluate the performance of this cycle (with fixed heat exchanger sizes) for specified flow rates and temperatures. This "Performance cycle" is used to make inferences regarding the behavior of the plant under varying operating conditions. Finally, the results of the Performance model are implemented in Fortran as a regression. This method is verified with the use of statistical regression analysis, as conferred later in this chapter.

Using a Rankine cycle with two feed-water heaters, pre-heat, and super-heat, a nominally 10*MW* cycle is constructed. The results of this analysis indicate that the Rankine cycle can successfully be modeled for a range of design points and equipment sizes using a single set of correlations. It should be noted that this model has been developed for only steam Rankine cycles with superheat, and has not been validated otherwise.

4.1 Design Rankine Cycle

The cycle under consideration is intended to accommodate the conversion of heat provided by the central receiver heat transfer fluid flow to electric power in the turbines with a combined output of 10 *MWe*. Although a 10 MWe system is considered in this analysis, the final results are scaled so the Rankine cycle model can be applied to any size cycle. The heat transfer fluid is supplied from the tower at 565°*C* under design conditions, while the cooling water is supplied to the condenser at 55°*F* (12.8°C), both at flow rates yet to be calculated. Figure 31 shows a simplified version of the design cycle configuration adapted from the Solar II power cycle schematic presented by Lippke (1995). The original figure as presented by Lippke is included in Appendix B.

A pre-heater, boiler, and super-heater handle the heat addition portion of the cycle. Each of these is modeled as a separate component since the heat transfer relationships that govern these three stages differ.

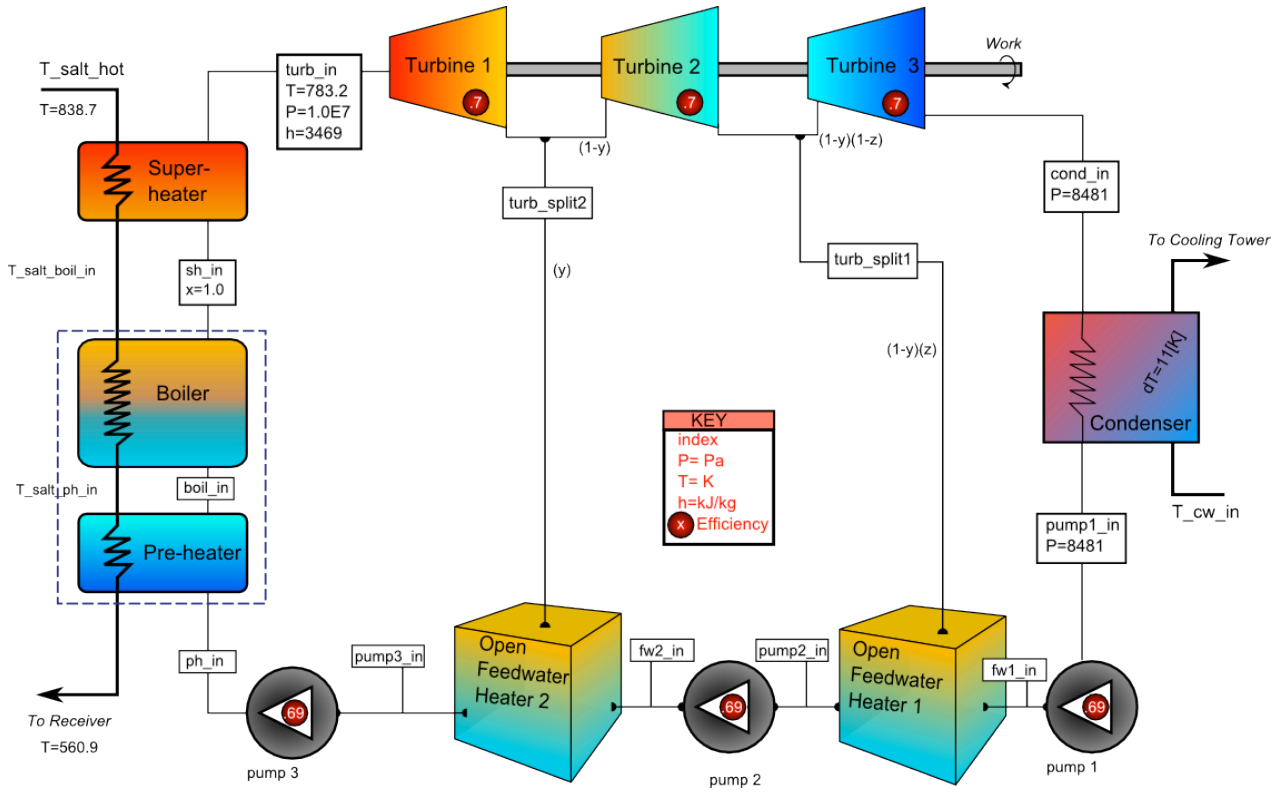


Figure 31: The Rankine cycle configuration used in the modeling of the power block. An open feedwater-heater is added, in addition to a super-heater.

4.1.1 A Rankine cycle thermodynamic analysis

To begin the analysis, a bookkeeping method is defined for each point of interest. Arrays containing the temperatures, pressures, enthalpies, and entropies will be denoted by the integers representing each of the points. In addition, the heat transfer fluid inlet temperature, outlet temperature, and type must be specified, and guess values for the pre-heater inlet temperature and boiler inlet temperature are provided. The steam mass flow rate can be specified is provided a guess value that will later be calculated, and the desired output power is specified.

```
"Define the bookeeping method"
    turb_in = 1                    "High-pressure turbine inlet"
    turb_split2 = 2                "The higher pressure splitter stage"
    turb_split1 = 3                "The lower pressure splitter stage"
    cond_in = 4                    "The condenser inlet"
    pump1_in = 5                   "The condenser outlet, pump inlet"
    fw1_in = 6                     "Low pressure feedwater heater inlet"
    pump2_in = 7                   "Intermediate stage pump inlet"
    fw2_in = 8                     "High pressure feedwater heater inlet"
```

```
    pump3_in = 9                          "High pressure pump inlet"
    ph_in = 10                            "Preheater HX inlet"
    boil_in = 11                          "Boiler inlet"
    sh_in = 12                            "Superheater HX inlet"
"Enter the state information"
"Salt conditions:"
    T_htf_hot = 838.7 [K]                 "The known salt hot inlet temperature"
    T_htf_boil_in = 774 [K]               "!guess value"
    T_htf_ph_in = 650 [K]                 "!guess value"
    T_htf_cold = 560.9 [K]                "The known salt temperature leaving the heat
exchangers"
"Mass flow of steam"
    m_dot_steam =11.8 [kg/s]               "! Guess value for Steam flow rate"
    w_dot_cycle = 10e6 [W]                "Desired power"
"Heat transfer fluid"
    HTF$='Salt (60% NaNO3, 40% KNO3)'
```

Typical turbine and pump efficiencies are specified, with the pump efficiencies equal to the value of an isentropic efficiency calculated from the Solar II cycle analyzed by Lippke (1995).

```
"Enter the turbine efficiencies"
eta_turb1 = .70 [-]
eta_turb2 = .70 [-]
eta_turb3 = .70 [-]                       "estimates"
"Enter the pump efficiencies"
eta_pump1 = .695 [-]                      "Calculated efficiencies from the Solar II design cycle"
eta_pump2 = .695 [-]
eta_pump3 = .695 [-]
```

The calculations for the cycle begin at the high pressure turbine inlet (*turb_in*), where desired pressure and temperature conditions are specified. The pressure at the turbine inlet is equal to the boiler pressure, since the pressure drop of the steam across the heat exchangers is neglected. The boiler pressure can be controlled in plant operation, and so this pressure is specified in the model.

```
"Start at the turbine 1 (highest pressure turbine) inlet"
T[turb_in] = 783.2 [K]
P[turb_in] = 1.0e7 [Pa]
h[turb_in] = enthalpy('Steam',T=T[turb_in], P=P[turb_in])
s[turb_in] = entropy('Steam',T=T[turb_in], P=P[turb_in])
```

The turbine splitter extraction phases can be fixed at a desired pressure to maximize cycle efficiency. The dependence of cycle efficiency on these pressures is relatively weak, and exact optimization is not needed. The pressure can be estimated as the saturation pressure of the temperature at the stage outlet. At this pressure, the corresponding

saturation temperature is calculated, and the intermediate splitter stage pressure is calculated by evenly dividing the difference in temperature between the first turbine inlet stage and the last turbine outlet stage into even increments (El Wakil, 1984). The resulting temperature at saturation is associated with the actual saturation pressure to be used at that splitter stage. This process is illustrated in Figure 32.

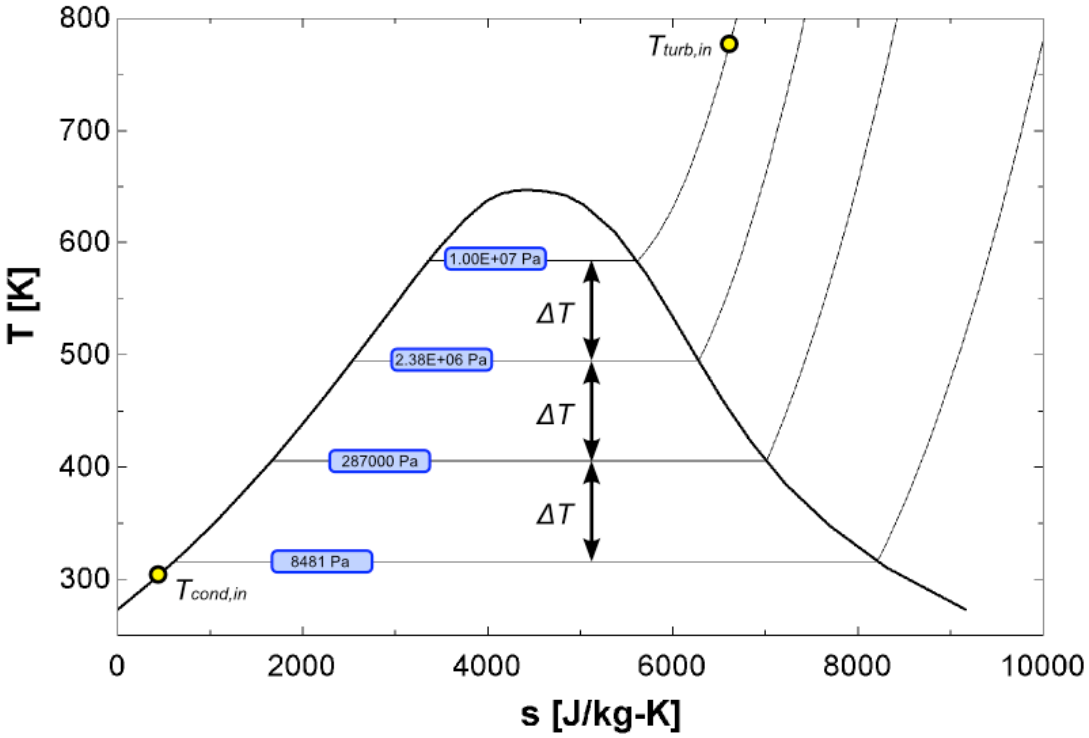


Figure 32: A temperature-entropy diagram for steam illustrating the pressure estimation in the turbine splitter stages. The pressure is estimated by dividing the temperature difference at saturation evenly between stages.

The saturation temperatures corresponding with the pressures at the inlet and outlet of the turbine stages are calculated, and require the specification of the condensing pressure, which is known for this model.

```
"Calculate saturation temps needed for equations"
P[cond_in] = 8481 [Pa]
TsatHI = T_sat('Steam',P=P[turb_in])
TsatLO = T_sat('Steam',P=P[cond_in])
```

The enthalpy at the outlet of the first turbine stage can be calculated using the known inlet conditions, the known pressure at the splitter stage, and the known isentropic efficiency.

```
"The turbine 1 outlet: turb_split2"
P[turb_split2]=P_sat('Steam',T=TsatHI-(TsatHI-TsatLO)/3)                "Pressure at the splitter stage"
h_s[turb_split2] = enthalpy('Steam',P=P[turb_split2],s=s[turb_in])      "The isentropic enthalpy"
h[turb_split2]=h[turb_in]+eta_turb1*(h_s[turb_split2]-h[turb_in])       "The enthlapy at outlet"
T[turb_split2] = temperature('Steam',P=P[turb_split2],h=h[turb_split2]) "Get the temperature"
s[turb_split2] = entropy('Steam',P=P[turb_split2],h=h[turb_split2])     "Get the entropy"
W_t1 = h[turb_in] - h[turb_split2]                                      "Specific work"
W_dot_t1 = W_t1*m_dot_steam                                             "Power from the first turbine"
```

At each of the two splitter stages, a portion of the steam mass flow rate is diverted into an open feed-water heater, as shown in the cycle schematic in Figure 31 above. The fractions that are extracted are denoted as y for the higher pressure feed-water heater and x for the lower pressure feed-water heater. The remaining fraction passes through the subsequent turbine stage(s) and condenser, with the total remaining fraction passing through the condenser being $(1-y)(1-x)$. To proceed with the analysis, guess values are provided for the extracted flow fractions. These values can later be adjusted to optimize the cycle efficiency. However, to initiate the calculations, the fraction are set to guess values that can be later replace with optimized values.

```
$ifnot minmax
    y = .11 [-]                 "!Optimized value for mass flow fraction to FWH 2"
    z = .1447 [-]               "!Optimized value for mass flow fraction to FWH 1"
$endif
```

In the same way that the high-pressure turbine outlet properties were calculated, the intermediate turbine outlet properties at the low-pressure feed-water heater can be evaluated. Whereas the outlet of the high-pressure turbine was fixed at a pressure representing the saturation temperature 1/3 of the way from *TsatHI* to *TsatLO*, the outlet of the intermediate turbine is evaluated at a saturation temperature 2/3 of the distance between these values.

```
"Turbine 2 outlet: turb_split1"
P[turb_split1]=P_sat('Steam',T=TsatHI-(TsatHI-TsatLO)*2/3)                    "Pressure at the splitter stage"
h_s[turb_split1] = enthalpy('Steam',P=P[turb_split1],s=s[turb_split2])  "Isentropic enthalpy at turbine inlet"
h[turb_split1] = h[turb_split2]+eta_turb2*(h_s[turb_split1]-h[turb_split2])    "Actualy enthalpy"
T[turb_split1] = temperature('Steam',P=P[turb_split1],h=h[turb_split1])    "Get the temperature"
s[turb_split1] = entropy('Steam',P=P[turb_split1],h=h[turb_split1])    "The entropy at the spiltter stage"
W_t2 = h[turb_split2] - h[turb_split1]                                    "Calculate the specific work"
W_dot_t2 = W_t2*m_dot_steam*(1-y)                                         "Calculate the power"
```

The condenser pressure at the outlet of the lowest pressure turbine has been specified in previous calculations, so the steam conditions at the condenser inlet are easily calculated. All of the turbine power rates are summed to provide the total power generated by the turbines.

```
"Turbine 3 outlet: cond_in"
h_s[cond_in] = enthalpy('Steam',P=P[cond_in],s=s[turb_split1])
h[cond_in] = h[turb_split1]+eta_turb3*(h_s[cond_in]-h[turb_split1])
T[cond_in] = temperature('steam',P=P[cond_in],h=h[cond_in])
W_t3 = h[turb_split1] - h[cond_in]
W_dot_t3 = W_t3*m_dot_steam*(1-y)*(1-z)
"--"
W_dot_tot = (W_dot_t1+W_dot_t2+W_dot_t3)*convert(W,MW)
```

The outlet condition of the steam is known to be saturated or slightly sub-cooled liquid. To maintain this condition, a temperature is specified such that the steam leaves as slightly sub-cooled liquid. The pressure at the outlet of the condenser is known since any pressure losses in the heat exchange equipment is neglected is neglected in this analysis.

```
"Condenser outlet:  pump1_in"
T[pump1_in] = T_sat('Steam',P=P[cond_in])-.5[K]      "Make sure the condenser is slightly subcooled"
P[pump1_in] = P[cond_in]                             "Assume no pressure drop across the condenser"
h[pump1_in] = enthalpy('Steam',T=T[pump1_in], P=P[pump1_in])        "Enthalpy"
s[pump1_in] = entropy('Steam',T=T[pump1_in], P=P[pump1_in])         "Entropy"
v[pump1_in] = volume('Steam',T=T[pump1_in], P=P[pump1_in])          "Specific volume"
    "Calculate condenser properties ----------"
    q_cond = h[cond_in] - h[pump1_in]                "Heat per unit mass"
    q_dot_cond = q_cond*m_dot_steam*(1-y)*(1-z)      "Heat rate"
```

The parasitic electric power required to run the pumps is virtually insignificant in the overall energy balance; however, it is likely of interest for sizing purposes. The following relationship is used to calculate the work done by the pump in compression of the working fluid:

$$W_p = v_{in} \cdot \left(\frac{p_{in} - p_{out}}{\eta_{pump}} \right) \tag{4.1}$$

The pump efficiency and the relationship presented in Eq. (4.1) provide enough information to calculate the work per-unit-mass done in compression by the low-pressure pump, and the actual outlet state of the steam from the pump. However, the fraction ($(1-y)x$) of the steam mass-flow that is extracted through the low-pressure feed-water heater is required to calculate the corresponding pump parasitic power. The pump power is scaled by this mass flow fraction, since the steam mass flow rate through the pump is decreased by the amount that is diverted before reaching the pump.

```
"Pump 1 outlet:   fw1_in"
P[fw1_in] = P[turb_split1]
W_p1=v[pump1_in]*(P[pump1_in]-P[fw1_in])/eta_pump1   "Work per unit mass of the pump"
W_dot_p1 = W_p1*m_dot_steam*(1-y)*(1-z)              "Power consumed by the pump"
"Calculate the change in enthalpy"
h[fw1_in]=h[pump1_in]-W_p1                           "energy balance on adiabatic pump"
    "Calculate the other values"
    T[fw1_in] = temperature('Steam',P=P[fw1_in],h=h[fw1_in])          "Temperature"
```

An energy balance on the low-pressure feed-water heater yields the enthalpy at the inlet of the intermediate pump. The pressure throughout the low-pressure feed-water system has been specified at the *turb_split1* state.

```
"Feedwater heater 1 outlet:  pump2_in"
h[pump2_in] = (1-z)*h[fw1_in] + z*h[turb_split1]          "Enthalpy balance"
P[pump2_in] = P[turb_split1]                              "No pressure drop through FWH"
T[pump2_in] = temperature('Steam',P=P[pump2_in],h=h[pump2_in])     "Get temperature"
v[pump2_in] = volume('Steam',P=P[pump2_in],h=h[pump2_in])          "Get volume"
```

The intermediate pump calculations are identical to the low-pressure pump, and another energy balance across the high-pressure feed-water heater is used to determine the conditions entering the high-pressure pump. High pressure pump conditions are calculated in the same way as the previous pumps, except that the boiler pressure is used as the steam pressure at the pump outlet.

```
"Pump 2 outlet:  fw2_in"
P[fw2_in] = P[turb_split2]                                  "Splitter stage pressure"
W_p2=v[pump2_in]*(P[pump2_in]-P[fw2_in])/eta_pump2          "Work per unit mass of the pump"
W_dot_p2 = W_p2*m_dot_steam*(1-y)                           "Pump power"
"Calculate change in enthalpy"
h[fw2_in] = h[pump2_in]-W_p2                                "Enthalpy at the feedwater inlet"
T[fw2_in] = temperature('Steam',P=P[fw2_in],h=h[fw2_in])    "Get temperature"
"Feedwater heater 2 outlet:  pump3_in"
h[pump3_in] = (1-y)*h[fw2_in]+y*h[turb_split2]              "Energy balance for HP FWH"
P[pump3_in] = P[turb_split2]                                "Pressure of the FWH system"
T[pump3_in] = temperature('Steam',P=P[pump3_in], h=h[pump3_in])      "Get temperature"
v[pump3_in] = volume('Steam',P=P[pump3_in],h=h[pump3_in])            "Get volume"
"Pump 3 outlet: ph_in"
P[ph_in] = P[turb_in]                                       "Boiler pressure"
W_p3 = v[pump3_in]*(P[pump3_in] - P[ph_in])/eta_pump3       "Work per mass"
W_dot_p3 = W_p3*m_dot_steam                                 "Pump power"
h[ph_in] = h[pump3_in]-W_p3                                 "Enthalpy at the pump outlet"
T[ph_in] = temperature('Steam',P=P[ph_in],h=h[ph_in])       "Get the temperature"
```

The following inventory of available information and modeling considerations for the heat-addition phase of the cycle is provided to clarify the calculation process. The heat-addition to the Rankine cycle occurs by means of a pre-heater heat exchanger, a steam boiler, and a super-heater heat exchanger. The pre-heater is placed at the low-temperature end of the heat transfer fluid flow, and at the low-energy end of the steam flow (following the high-pressure pump). The pre-heater provides thermal energy to the sub-cooled steam such that the steam flow leaving the pre-heater under design conditions is at saturated liquid conditions.

Physically, the pre-heater and boiler may exist as the same piece of equipment. However, the heat-transfer correlations that apply to sub-cooled liquid heat transfer and the correlations that apply to heat transferred with one side undergoing a phase change (the steam in the boiler) are different, and so must be modeled differently.

The super-heater heat exchanger is also modeled as a separate piece of equipment. Steam generated in the boiler is extracted into the super-heater where heat transfer between the steam vapor and the hottest portion of the heat transfer fluid takes place.

Therefore, from the preceding discussion, from previous calculations, and from the design diagram in Figure 31, the following conditions are known:

- Pre-heater inlet steam condition

- Pre-heater outlet / boiler inlet steam quality ($\chi = 0$)
- Pre-heater outlet heat transfer fluid temperature ($T_{htf,cold}$)
- Boiler outlet / super-heater inlet steam condition
- Super-heater heat transfer fluid inlet temperature ($T_{htf,hot}$)
- Super-heater outlet / high-pressure turbine inlet steam condition
- Steam mass flow rate

Among the unknown information in the heat-addition portion of the cycle are the following parameters:

- Heat transfer fluid mass flow rate ($\dot{m}_{htf}$)
- Heat transfer fluid boiler inlet temperature ($T_{htf,boil,in}$)
- Heat transfer fluid pre-heater inlet temperature ($T_{htf,ph,in}$)
- Pre-heater, boiler, and super-heater conductance (UA_{ph}, UA_{boil}, UA_{sh}, respectively)

To proceed with the analysis, the pre-heater outlet / boiler inlet conditions are calculated with known quality and pressure. The heat flow in the pre-heater can then be determined, and the heat transfer rate follows.

```
"preheater outlet: boil_in"
x[boil_in] = 0 [-]                                          "Fixed quality at boiler inlet"
P[boil_in] = P[turb_in]                                     "No pressure drop"
T[boil_in] = temperature('Steam',P=P[boil_in],x=x[boil_in]) "Get temperature"
h[boil_in] = enthalpy('Steam',P=P[boil_in],x=x[boil_in])    "Get enthalpy"
    "Calculate the preheater size"
    q_ph = h[boil_in] - h[ph_in]                            "Pre-heater heat flow"
    q_dot_ph = q_ph*m_dot_steam                             "Heat flow rate"
    "--"
    c_w_ph = specheat('Steam',T=((T[ph_in]+T[boil_in])/2),P=P[boil_in]) "PH steam spec.heat"
    c_dot_w_ph = c_w_ph*m_dot_steam                         "PH steam capacitance rate"
```

Since the heat transfer fluid conditions at the pre-heater are needed to determine the conductance of the heat exchanger, the mass flow rate of the heat transfer fluid should be calculated at this point. With the known pre-heater inlet steam enthalpy and the known turbine inlet steam enthalpy, the total heat flow per-unit-mass into the steam is calculated. The mass flow rate of the heat transfer fluid can then be obtained by employing the energy balance shown in Eq. (4.2) below, where $c_{htf,ave}$ is evaluated at the average heat

transfer fluid temperature in the cycle. This energy balance assumes that there are no thermal losses from the jacket of the boiler heat exchangers.

$$\dot{m}_{htf} = \frac{\left(h_{turb,in} - h_{ph,in}\right)\cdot \dot{m}_{steam}}{c_{htf,ave}\cdot\left(T_{htf,hot} - T_{htf,cold}\right)} \tag{4.2}$$

```
"---Calculate the HTF flow rate before proceeding"
m_dot_htf = (h[turb_in]-h[ph_in])*m_dot_steam/(c_(HTF$,(T_htf_hot +T_htf_cold)/2)*(T_htf_hot-
T_htf_cold))
"---"
c_htf_ph = c_(HTF$,((T_htf_cold+T_htf_ph_in)/2))          "PH HTF spec.heat"
c_dot_htf_ph = c_htf_ph*m_dot_htf                          "PH HTF cap rate"
```

With all of the information about the flows passing through the pre-heater defined, the heat exchanger size can be determined. The Effectiveness-NTU method is the most convenient for characterizing the simple heat exchanger models used in this evaluation. The heat-exchanger effectiveness (ε) is defined as in Eq. (4.3) such that it indicates the fraction of heat exchanged between the hot and cold streams compared to the maximum heat flow possible.

$$\dot{q} = \varepsilon \dot{q}_{\max} \tag{4.3}$$

The maximum heat flow is defined in terms of the capacitance rate and the maximum temperature difference between hot and cold side streams as shown in Eq. (4.4).

$$\begin{aligned} &\dot{q}_{\max} = \dot{C}_{\min}\left(T_{H,in} - T_{C,in}\right) \\ &where: \\ &\dot{C}_{\min} = \min\left[\left(c_{p,steam}\cdot\dot{m}_{steam}\right),\left(c_{cw}\cdot\dot{m}_{cw}\right)\right] \end{aligned} \tag{4.4}$$

Since the specific heat of a fluid undergoing a phase change (as in the condenser or boiler) is effectively infinite, the minimum capacitance rate for the heat transfer relationship in these components will always be on the non-steam side. Combining Eqs. (4.3) and (4.4), we get a useful relationship between the effectiveness, stream inlet temperatures, and the heat transfer in the heat exchanger.

$$\dot{q} = \varepsilon \cdot \dot{C}_{\min} \left(T_{H,in} - T_{C,in} \right) \tag{4.5}$$

The number of transfer units (NTU) represents the dimensionless size of the heat exchanger. The UA associated with the particular heat exchanger is scaled by the minimum capacitance rate described in Eq. (4.4).

$$NTU = \frac{UA}{\dot{C}_{\min}} \tag{4.6}$$

A final consideration in calculating the heat exchanger size is the relationship between effectiveness and *NTU*, which can be determined based on the heat exchanger configuration and the capacitance ratio. This ratio is defined as the minimum capacitance rate divided by the maximum capacitance rate. For streams that are undergoing a phase change, the effective specific heat of the fluid is infinite since the energy flow is latent heat. Therefore, the maximum capacitance rate is infinite, and the ratio (C_R) is zero. For this situation, the *NTU* can be calculated as in Eq. (4.7), and is independent of heat exchanger configuration (Nellis & Klein, 2009).

$$NTU = -\ln\left(1 - \varepsilon\right) \tag{4.7}$$

For heat exchangers where both flows are single-phase, the relationship between *NTU* and effectiveness depends on the configuration of the heat exchanger. In this analysis, a counter-flow heat exchanger is used. The *NTU* is calculated with the following relationship (Nellis & Klein, 2009):

$$NTU_{PH} = \frac{\ln\left(\dfrac{\varepsilon_{ph} - 1}{\varepsilon_{ph} C_{R,PH} - 1} \right)}{C_{R,PH} - 1} \tag{4.8}$$

The pre-heater capacitance ratio is calculated as follows:

```
"Now use the effectiveness-NTU relationship to calculate the heat exchanger size"
c_dot_min_ph = MIN(c_dot_w_ph, c_dot_htf_ph)
c_dot_max_ph = MAX(c_dot_w_ph, c_dot_htf_ph)
C_R_PH = C_dot_min_ph/C_dot_max_ph                "The capacitance ratio"
```

At this point, enough information has been calculated to compute the heat transfer fluid pre-heater inlet temperature that was previously assigned a guess value.

```
"The energy balance on the hot side of the HX"
T_htf_ph_in = T_htf_cold + q_dot_ph/c_dot_htf_ph          "Calculate the inlet htf temperature, NOTE:
negates previous guess temp"
```

Now the pre-heater heat exchanger conductance can be computed using the relationship between the *NTU* and effectiveness shown in Eq. (4.8). This relationship is implemented in the EES code through a call to the '*HX*' library. The pre-heater *UA* is then calculated.

```
"The effectiveness-NTU relationship for countercurrent HX implemented"
q_dot_ph = epsilon_ph*C_dot_min_ph*(T_htf_ph_in - T[ph_in])              "The preheater heat flow"
NTU_ph=HX('counterflow', epsilon_ph, C_dot_min_ph, C_dot_max_ph, 'NTU')  "The NTU for the
preheater"
UA_ph = NTU_ph*C_dot_min_ph                                              "The corresponding UA"
```

With a previously calculated heat transfer fluid mass flow rate and pre-heater inlet temperature, as well as known steam conditions, the boiler conductance calculation is relatively straight-forward. Figure 33 shows an energy balance on the boiler. Note the effectively infinite specific heat capacity of the steam flow that results from the latent heat addition through a phase change from liquid to vapor.

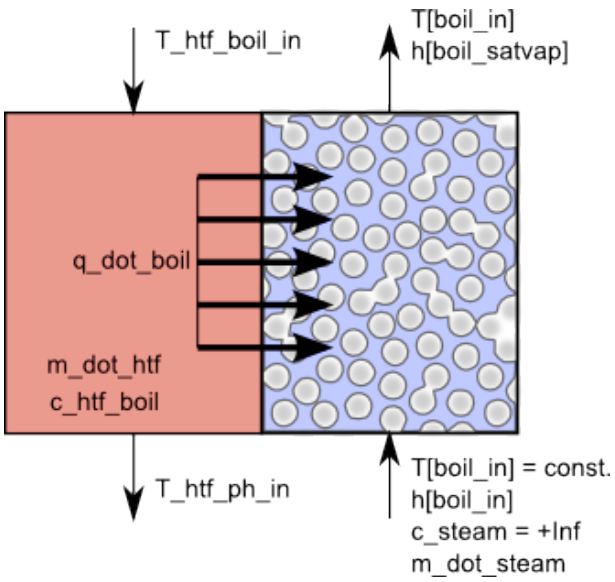


Figure 33: An energy balance representing the heat flow across the boiler from the heat transfer fluid to the boiling steam flow.

The effectively infinite capacitance rate on the steam side of the heat exchanger requires the use of the effectiveness-*NTU* relationship shown above in Eq. (4.7). The heat flow in the pre-heater is calculated with the steam inlet and outlet enthalpies on a per-unit-mass basis as q_{boil}. The heat transfer rate can then be determined as the product of the heat per-unit-mass by the steam mass flow rate. The heat transfer fluid boiler inlet temperature can be determined with an energy balance on the HTF side of the boiler heat exchanger.

```
"boiler outlet:  sh_in"
x[sh_in] = 1 [-]
P[sh_in] = P[boil_in]
T[sh_in] = temperature('Steam', P=P[sh_in],x=x[sh_in])
h[sh_in] = enthalpy('Steam',P=P[sh_in], x=x[sh_in])
    "Calculate the boiler size"
    q_boil = h[sh_in] - h[boil_in]
    q_dot_boil = q_boil*m_dot_steam
    "--"
    c_htf_boil = c_(HTF$, (T_htf_boil_in + T_htf_ph_in)/2)
    c_dot_htf_boil = c_htf_boil*m_dot_htf
    q_dot_boil = epsilon_boil*c_dot_htf_boil*(T_htf_boil_in - T[boil_in])
    NTU_boil = -ln(1-epsilon_boil)                    "The boiler NTU"
    UA_boil = NTU_boil*m_dot_htf*c_htf_boil           "The corresponding UA"
    "Calculate htf boiler inlet temp, replaces guess value"
    T_htf_boil_in = T_htf_ph_in + q_dot_boil/c_dot_htf_boil
```

The final calculation in the heat-addition portion of the Rankine cycle is the computation of the super-heater heat exchanger conductance. This heat exchanger is similar to the

pre-heater in configuration and in analysis with the relationship shown previously in Eq. (4.8) also applying to the super-heater. Once the super-heater heat flow has been calculated, the unknown heat transfer fluid boiler inlet temperature is established, and the previously specified guess temperature is removed.

```
"Calculations for the superheater"
q_sh = h[turb_in]-h[sh_in]                                    "Heat flow in the super-heater"
q_dot_sh = q_sh*m_dot_steam                                   "Heat flow rate"
c_w_sh = specheat('Steam',T=((T[sh_in]+T[turb_in])/2), P=P[turb_in])      "Spec.heat steam"
c_dot_w_sh = c_w_sh*m_dot_steam                               "Cap rate steam"
c_htf_sh = c_(HTF$, (T_htf_hot + T_htf_boil_in)/2)            "Spec heat HTF"
c_dot_htf_sh = c_htf_sh*m_dot_htf                             "Cap rate HTF"
c_dot_min_sh = MIN(c_dot_w_sh, c_dot_htf_sh)                  "Minimum cap rate"
c_dot_max_sh = MAX(c_dot_w_sh, c_dot_htf_sh)                  "Maximum cap rate"
C_R_sh = c_dot_min_sh/c_dot_max_sh                            "Cap ratio"
epsilon_sh = q_dot_sh/(c_dot_min_sh*(T_htf_hot-T[sh_in]))                 "effectiveness"
NTU_sh = HX('counterflow',epsilon_sh, c_dot_min_sh, c_dot_max_sh, 'NTU')
UA_sh = NTU_sh*c_dot_min_sh                                   "Conductance"
```

The only remaining design-cycle computations are the conductance of the condenser and the flow rate of the cooling water through the condenser. For the design case, both the cooling water inlet and outlet temperatures must be specified. Expressed in terms of the difference between the saturation temperature and the cooling water circulating temperature, the recommended temperature range for the inlet temperature is between 20 and 30°F (11 to 17°C) below saturation temperature (El-Wakil, 1984). The difference between the outlet temperature should not be less than about 5°F (2.8°C). The actual temperature can be calculated according to:

$$T_{cw} = T_{sat}\left('Steam', P = P_{cond,in}\right) - \Delta T \tag{4.9}$$

where ΔT represents the specified temperature difference between the saturation temperature and the cooling water temperature.

```
"The condenser is modeled as a HX with 1 side of infinite cap. rate"
    "Calculate the cooling water inlet/outlet temps (El-Wakil, M.M., Powerplant Technology, 1984) "
    dT_in = 15 [K]                                            "Difference between sat. temp and inlet
temp"
    T_cw_in = T_sat('Steam',P=P[cond_in]) - dT_in             "The cooling water inlet temp."
    dT_out= 4 [K]                                             "Difference between sat. temp and outlet
temp"
    T_cw_out = T_sat('Steam',P=P[cond_in]) - dT_out           "The cooling water outlet temp."
    T_cwinF = converttemp(K,F,T_cw_in)                        "Temp in F"
```

```
T_cwoutF = converttemp(K,F,T_cw_out)                    "Temp in F"
```

The cooling water temperatures are thus calculated with an inlet temperature difference specified to be 15*K*, and an outlet temperature difference specified to be 4*K*, yielding actual inlet and outlet temperatures of 81.7°F and 101.5°F, respectively (27.6°C and 38.6°C). Finally, the condenser conductance can be determined.

```
c_p_cw = specheat('Water',T=T_cw_in,P=101000[Pa])   "The specific heat of the cooling water"
m_dot_cw = q_dot_cond/(c_p_cw*(T_cw_out - T_cw_in))  "Calculate the cooling water flow rate"
q_dot_cond = epsilon_cond*m_dot_cw*c_p_cw*(T[cond_in]-T_cw_in)    "Effectiveness"
NTU_cond = -ln(1-epsilon_cond)                       "The transfer units for this heat exchanger"
UA_cond = NTU_cond*m_dot_cw*c_p_cw                   "Conductance"
```

To summarize the cycle performance, the total cycle heat addition, total cycle work minus parasitics, and the total thermal efficiency are determined.

```
"Total heat addition"
q_dot_hot_tot = q_dot_ph+q_dot_boil+q_dot_sh
"Calculate the total work from the cycle"
W_dot_cycle = W_dot_t1 + W_dot_t2 + W_dot_t3 + W_dot_p1 + W_dot_p2 + W_dot_p3
"Calculate the total thermal efficiency of the cycle"
eta_cycle = W_dot_cycle/q_dot_hot_tot
```

Cycle parameters such as heat exchanger conductances and fluid mass flow rates that are determined in the design phase can now be used in building the performance cycle model. Table 13 summarizes the noteworthy cycle output, and also indicates whether each particular item is set as a parameter in the performance Rankine cycle model.

Table 13: Output from the Design cycle. The "Set" column indicates whether the item is set as a parameter in the performance cycle model.

PARAMETER	DESIGN VALUE	DESCRIPTION	SET?
$\dot{m}_{HTF}$	71.43 [kg/s]	The mass flow rate of the salt	◀
$\dot{m}_{cw}$	438.1 [kg/s]	The mass flow rate of the cooling water in the condenser	◀
$\dot{m}_{steam}$	11.84 [kg/s]	The reference steam mass flow rate (an input	
UA_{boil}	169,947 [W/K]	The boiler conductance	◀
UA_{cond}	2.422×10^{6} [W/K]	The condenser conductance	◀
UA_{PH}	113,967 [W/K]	The pre-heater conductance	◀
UA_{SH}	95,698 [W/K]	The super-heater conductance	◀
$\dot{W}_{cycle}$	10.0×10^{6} [W]	The cycle electric power output	
η_{cycle}	0.2726 [-]	The cycle thermal efficiency	
$T_{htf,boil,in}$	765.8 [K]	The heat transfer fluid boiler inlet temp	
$T_{htf,ph,in}$	622.1 [K]	The heat transfer fluid pre-heater inlet temp	

The Rankine cycle temperature-entropy diagram is shown at design conditions in Figure 34.

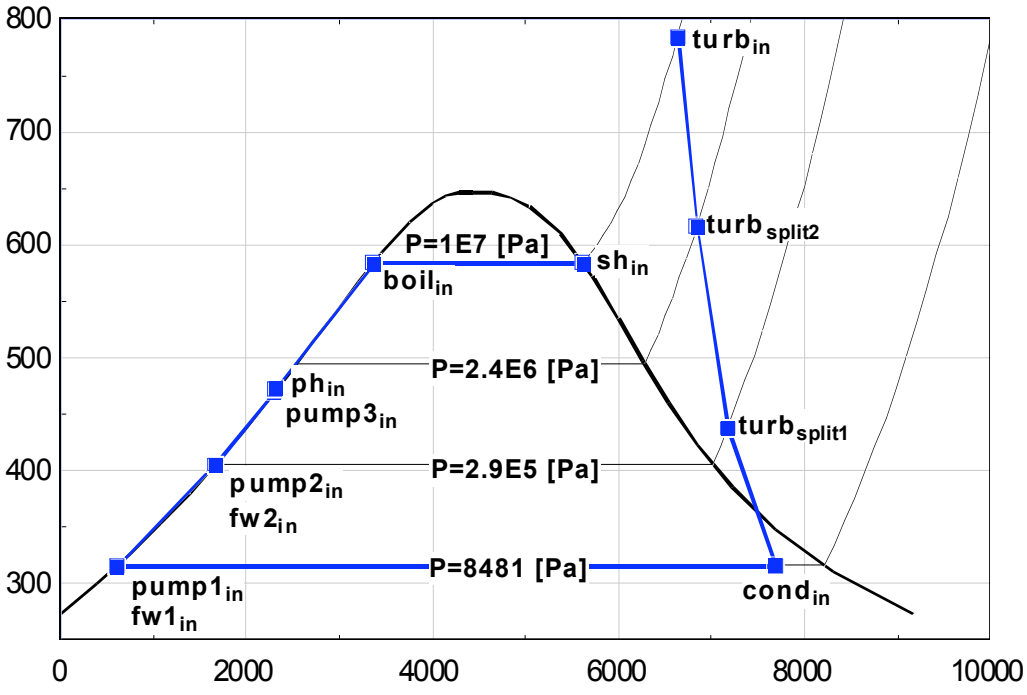


Figure 34: The Temperature-Entropy diagram for steam, with the Rankine cycle under consideration overlaid.

4.2 Performance Rankine Cycle

The previous analysis of the Rankine cycle determined the heat exchanger conductance and mass flow rates needed to match the specified nominal power output and thermal resources of the cycle. In the following analysis, the conductance of the heat exchangers will be fixed at the design value, and the mass flow rates of the cooling water in the condenser and the salt in the boiler/super-heater will also be specified as model inputs. However, the power output from the cycle and the mass flow rate of the steam is adjusted with the varying temperature of the hot heat transfer fluid. This section walks through the analysis with these new constraints, and highlights the conceptual differences with respect to the previous design analysis.

As the steam mass flow rate through the turbines varies, the isentropic efficiency of the turbine also varies. To compensate for this, the addition of a varying mass flow rate requires an adjustment of the steam turbine isentropic efficiency. This efficiency is de-rated to accommodate the throttling of the mass flow rate according to its deviation from the nominal – or reference – flow conditions. A function is added to the EES model that calculates the de-rate factor for each turbine stage, according to the relationship in Eq. (4.10) (Patnode, 2005).

$$\eta_{red} = \left[0.191 - 0.409 \frac{\dot{m}}{\dot{m}_{ref}} + 0.218 \left(\frac{\dot{m}}{\dot{m}_{ref}} \right)^2 \right] \tag{4.10}$$

This value is multiplied by the turbine efficiency at nominal steam mass flow rate as shown in Eq. (4.11).

$$\eta_{turb} = \left(1 - \eta_{red}\right) \cdot \eta_{turb,ref} \tag{4.11}$$

The relationship is implemented in the EES program as a function, which is shown here.

```
"This function calculates the reduction in turbine efficiency based on the throttling of the mass
flow rate of steam through the turbine"
function eta_red(m_dot, m_dot_ref)
        "This function calculates the reduction in turbine efficiency based on the throttling of the mass
        flow rate of steam through the turbine"
        if(m_dot<m_dot_ref) then
                pc_red:=.191 - .409*(m_dot/m_dot_ref) + .218*(m_dot/m_dot_ref)^2
        else
                eta_red:=0      "The correlation only applies to throttling, and not a mass flow rate
                                greater than the reference"
        endif
end function
```

In the design cycle, the size of the heat exchangers was calculated to match the cycle output and temperature requirements. The *UA* values for the condenser, boiler, and super-heater heat-exchangers are now set in the performance calculations. Additionally, the mass flow rates of the cooling water in the condenser and the heat transfer fluid in the boiler and super-heater are specified to match the requirement of 10*MWe* output at nominal conditions. Table 14 documents the full set of constraints and calculated outputs for both the design and performance cycle models.

Table 14: Summary of variables and fixed parameters in the performance and design phase Rankine cycle models

	INPUT PARAMETERS	CALCULATED OUTPUT
Design model	• Steam mass flow rate • Steam conditions throughout cycle • Boiler pressure • Condenser cooling water inlet temp, outlet temp • Heat transfer fluid inlet, outlet, temperatures	• Heat transfer fluid mass flow rate • Pre-heater, boiler, super-heater, and condenser conductance • Heat transfer fluid boiler and pre-heater inlet temperatures • Condenser circulating water mass flow rate • High and low pressure turbine efficiencies • High and low pressure pump efficiencies • Cycle power, thermal efficiency
Performance model	• Pre-heater, boiler, super-heater, and condenser conductance • Heat transfer fluid mass flow rate • Heat transfer fluid inlet temperature • Condenser cooling water inlet temp • Condenser cooling water mass flow • Pump and turbine efficiency • Boiler pressure	• Steam conditions throughout the cycle • Steam mass flow rate • Condenser cooling water outlet temperature • Heat transfer fluid boiler inlet, outlet temps • Cycle power, thermal efficiency

The performance model input parameter set changes the nature of the cycle analysis so that the model reflects more realistic Rankine cycle behavior. The following section deals with building new component relationships to accommodate these constraints.

4.2.1 The Rankine Cycle model under non-design operation

Under non-design conditions, the Rankine cycle is subject to several varying inputs; the hot salt flow rate, hot salt inlet temperature, condenser cooling water inlet temperature (as a function of ambient wet-bulb temperature), and cooling water flow rate can all potentially vary during operation.

The heat exchanger conductance values are anchored at the specified value under nominal conditions, but do vary in relation to the steam mass flow rate. This variation is accounted for by scaling the condenser *UA* with respect to the proportion of the fluid mass flow rates to their reference mass flow rate at nominal conditions. This relationship is shown in Eq. (4.12) (Patnode, 2006).

$$\frac{UA}{UA_{ref}} = \left(\frac{\dot{m}_1^{0.8} \cdot \dot{m}_2^{0.8}}{\dot{m}_{1,ref}^{0.8} \cdot \dot{m}_{2,ref}^{0.8}} \right) \left(\frac{\dot{m}_{1,ref}^{0.8} + \dot{m}_{2,ref}^{0.8}}{\dot{m}_1^{0.8} + \dot{m}_2^{0.8}} \right) \tag{4.12}$$

where: $\dot{m}_1$ is the first stream mass flow rate

$\dot{m}_{1,ref}$ is the first stream reference mass flow rate

$\dot{m}_2$ is the second stream mass flow rate

$\dot{m}_{2,ref}$ is the second stream reference mass flow rate

In the case where the first mass flow rate and second mass flow rate remain in the same proportion at partial load conditions to the reference case, the relationship can be simplified (Patnode, 2006). This is the case in this model, and so the relationship in Eq. (4.13) can be used.

$$UA = UA_{ref} \left(\frac{\dot{m}_1}{\dot{m}_{1,ref}} \right)^{0.8} \tag{4.13}$$

To begin the analysis, the reference values and conditions calculated in the design Rankine cycle can be specified, as well as the same bookkeeping method as the design cycle analysis.

```
"Sizing information from design calculations"
P_boil=1.e7 [Pa]                          "Boiler pressure is fixed"
m_dot_ref= 12.27 [kg/s]                   "Reference steam mass flow rate"
m_dot_HTF_ref = 73.36 [kg/s]              "Reference HTF mass flow rate"

UA_boil_ref=169964 [W/K]                  "Reference boiler conductance"
UA_cond_ref=2422022 [W/K]                 "Reference condenser conductance"
UA_ph_ref=113967 [W/K]                    "Reference pre-heater conductance"
UA_sh_ref=95698 [W/K]                     "Reference super-heater conductance"

eta_pump1= .695 [-]                       "Reference low-pressure pump efficiency"
eta_pump2= .695 [-]                       "Reference intermediate pump efficiency"
eta_pump3= .695 [-]                       "Reference high-pressure pump efficiency"
eta_turb1_ref= .7[-]                      "Reference low-pressure turbine efficiency"
eta_turb2_ref= .7[-]                      "Reference intermediate turbine efficiency"
eta_turb3_ref= .7[-]                      "Reference high-pressure turbine efficiency"
HTF$='Salt (60% NaNO3, 40% KNO3)'
```

```
"Define the bookeeping method"
    turb_in = 1                           "High-pressure turbine inlet"
    turb_split2 = 2                       "The higher pressure splitter stage"
    turb_split1 = 3                       "The lower pressure splitter stage"
    cond_in = 4                           "The condenser inlet"
    pump1_in = 5                          "The condenser outlet, pump inlet"
    fw1_in = 6                            "Low pressure feedwater heater inlet"
    pump2_in = 7                          "Intermediate stage pump inlet"
    fw2_in = 8                            "High pressure feedwater heater inlet"
    pump3_in = 9                          "High pressure pump inlet"
    ph_in = 10                            "Preheater HX inlet"
    boil_in = 11                          "Boiler inlet"
    sh_in = 12                            "Superheater HX inlet"
```

One function of the performance Rankine cycle model is the prediction of cycle behavior under changing operating conditions. The varying conditions in question are manifested in either the heat addition or heat removal phases of the cycle, and can be accounted for in variation of the heat transfer fluid inlet temperature and mass flow rate, and in the condenser cooling water inlet temperature and mass flow rate. Initial values of these parameters are chosen, with the possibility that they will later be removed for use in parametric studies.

```
    T_HTF_hot = 838.7 [K]          "The temperature of the HTF coming from the hot storage tank"
    m_dot_cw= 438.1 [kg/s]
    m_dot_htf= 73.36 [kg/s]
    T_dp_F = 81.69 [F]             "The dewpoint temperature, the cw inlet temp closely tracks this
value"
```

A number of guess values will be required for this analysis. These values provide temporary information until the actual value can be determined with available information. The guess values are set using the results from the Design cycle model.

```
T[cond_in]=315.8 [K]                "!Guess temp for condenser inlet"
P[cond_in]=8481 [Pa]                "!Guess condensing pressure"
m_dot_steam = m_dot_ref             "!Guess steam mass flow rate"
y=.11                               "!Guess extraction fraction y"
z=.1447                             "!Guess extraction fraction z"
T_htf_boil_in=765[K]                "!Guess boiler salt inlet temp"
T_htf_ph_in=622[K]                  "!Guess preheater salt inlet temp"
T_htf_cold = 561[K]                 "!Guess salt outlet temp"
T[turb_in]=783 [K]                  "!Guess temp"
T[boil_in] = 584[K]                 "!Guess temp"
T[sh_in] = 585[K]                   "!Guess"
```

The most convenient place to begin the cycle calculations is at the inlet to the high pressure turbine. At this position, the steam pressure is known and the steam temperature can be closely approximated. Steam conditions are thus specified.

```
"Start with the turbine"
P[turb_in]=P_boil
h[turb_in] = enthalpy('Steam',T=T[turb_in],P=P[turb_in])
s[turb_in] = entropy('Steam',T=T[turb_in],P=P[turb_in])
```

The saturation temperatures for the high and low pressures surrounding the turbine stages are calculated to assist in determining the pressure at the splitter stages (for more discussion on this method, see the Performance cycle model, especially Figure 32).

```
"Calculate saturation temps needed for equations"
TsatHI = T_sat('Steam',P=P[turb_in])
TsatLO = T_sat('Steam',P=P[cond_in])
```

The power provided by the first turbine stage is calculated by considering the pressure at the outlet of this stage, which is the saturation pressure of a temperature 1/3 of the way from *TsatHI* to *TsatLO*. This, in combination with an isentropic efficiency relationship, is used to calculate the work output per-unit-mass of the turbine and the outlet conditions of the first turbine. The power output is simply the work multiplied by the steam mass flow rate.

```
"Turbine 1 outlet: turb_split2"
```

```
P[turb_split2]=P_sat('Steam',T=TsatHI-(TsatHI-TsatLO)/3)
h_s[turb_split2]=enthalpy('Steam',P=P[turb_split2],s=s[turb_in])
eta_turb1= eta_turb1_ref*(1-eta_red(m_dot_steam, m_dot_ref))
h[turb_split2] = h[turb_in]+eta_turb1*(h_s[turb_split2]-h[turb_in])
T[turb_split2] = temperature('Steam',P=P[turb_split2],h=h[turb_split2])
s[turb_split2] = entropy('Steam',P=P[turb_split2],h=h[turb_split2])
"Work"
W_t1 = h[turb_in]-h[turb_split2]
W_dot_t1 = W_t1*m_dot_steam
```

The intermediate stage turbine calculations mimic the high-pressure turbine, except the outlet pressure is now 2/3 the distance between *TsatHI* and *TsatLO*, just as in the Design model. The pressure of the splitter stages is allowed to adjust to maintain this relationship throughout operation. Also, since the mass flow rate of the steam through the second turbine is reduced by the proportion of steam that split off into the feed-water heater (*y*), the power produced by the turbine is diminished by the fraction (*1-y*).

```
"Turbine 2 outlet: turb_split1"
P[turb_split1]=P_sat('Steam',T=TsatHI-(TsatHI-TsatLO)*2/3)
h_s[turb_split1]=enthalpy('Steam',P=P[turb_split1],s=s[turb_split2])
eta_turb2 = eta_turb2_ref*(1-eta_red(m_dot_steam,m_dot_ref))
h[turb_split1] = h[turb_split2]+eta_turb2*(h_s[turb_split1]-h[turb_split2])
T[turb_split1] = temperature('Steam',P=P[turb_split1],h=h[turb_split1])
s[turb_split1] = entropy('Steam',P=P[turb_split1],h=h[turb_split1])
"Work"
W_t2 = h[turb_split2]-h[turb_split1]
W_dot_t2 = W_t2*m_dot_steam*(1-y)
```

The low-pressure turbine stage is also calculated in the same way as the previous turbine stages. The guess temperature that has been used for the condenser inlet stage can be removed as the steam conditions are determined, and the total work provided by the turbine stages is summed under the value *W_dot_tot*.

```
"Turbine 3 outlet: cond_in"
h_s[cond_in] = enthalpy('Steam',P=P[cond_in],s=s[turb_split1])
eta_turb3 = eta_turb3_ref*(1-eta_red(m_dot_steam,m_dot_ref))
h[cond_in] = h[turb_split1]+eta_turb3*(h_s[cond_in]-h[turb_split1])
T[cond_in] = temperature('Steam',P=P[cond_in],h=h[cond_in])
"Work"
W_t3 = h[turb_split1]-h[cond_in]
W_dot_t3 = W_t3*m_dot_steam*(1-y)*(1-z)
W_dot_tot = W_dot_t1+W_dot_t2 + W_dot_t3
```

In the condenser, the mass flow rate of the cooling water, the cooling water inlet temperature, and the conductance of the condenser heat-exchanger have been specified.

Since the mass flow rate of the steam is a function only of activity in the boiler, the condenser steam mass flow rate is also known.

Determining the condenser outlet conditions is somewhat uncomplicated once the cooling water inlet temperature, mass flow rate and specific heat are provided. The *NTU* and effectiveness are then found using the *UA*, as has been previously demonstrated.

The effectiveness relationship is used to calculate the heat flow from the condensing steam, and energy balances on the steam flow and cooling water flow help establish the outlet water enthalpy and the cooling water outlet temperature.

```
"The condenser"
T_cw_in = converttemp(F,K,T_dp_F)
c_p_cw = specheat('Steam',T=T_cw_in,P=101300[Pa])
c_dot_cw = m_dot_cw*c_p_cw
"--"
UA_cond=UA_cond_ref*(m_dot_steam/m_dot_ref)^(.8)
NTU_cond = UA_cond/c_dot_cw
epsilon_cond = 1-exp(-NTU_cond)
q_dot_cond = epsilon_cond*c_dot_cw*(T[cond_in]-T_cw_in)
q_cond = q_dot_cond/(m_dot_steam*(1-y)*(1-z))
T_cw_out = T_cw_in + q_dot_cond/(c_dot_cw)
```

Condenser outlet conditions are calculated using the condenser inlet enthalpy and the heat flow across the condenser heat exchanger (*q_cond*). The quality of the steam exiting the condenser is known to be saturated liquid (or slightly sub-cooled). Specification of this quality removes the condenser inlet steam temperature guess value that was provided. The condenser pressure can then be back-calculated from other known properties.

```
"Condenser outlet: pump1_in"
h[pump1_in] = h[cond_in] - q_cond
P[pump1_in] = P[cond_in]
T[pump1_in] = temperature('Steam',P=P[pump1_in],x=x[pump1_in])
v[pump1_in] = volume('Steam',P=P[pump1_in],x=x[pump1_in])
"Since the condenser outlet must be at least at a quality of zero, use that as a setpoint"
x[pump1_in] = 0. [-]
P[cond_in] = pressure('Steam',h=h[pump1_in],x=x[pump1_in])          "Replaces guess value"
```

Standard pump calculations are used to fix the steam condition at the low-pressure feed-water inlet. Since the quality of the steam exiting the feed-water system must be saturated liquid or slightly sub-cooled, a quality of zero is used. Fixing this state provides enough

information to remove the guess value for the flow rate fraction diverted through the low-pressure feed-water system (z). To convert the work per-unit-mass to power, the work must be multiplied by the steam mass flow rate, which has not yet been determined. For the time being, the reference steam mass flow rate can be used as a guess value.

```
"Pump1 outlet: fw1_in"
P[fw1_in]=P[turb_split1]
W_p1= v[pump1_in]*(P[pump1_in]-P[fw1_in])/eta_pump1
h[fw1_in] = h[pump1_in]-W_p1
W_dot_p1 = W_p1*m_dot_steam*(1-y)*(1-z)
T[fw1_in] = temperature('Steam',P=P[fw1_in],h=h[fw1_in])

"Feedwater 1 outlet: pump2_in"
P[pump2_in] = P[turb_split1]
x[pump2_in] = 0[-]
h[pump2_in] = enthalpy('Steam',P=P[pump2_in],x=x[pump2_in])
h[pump2_in]=(1-z)*h[fw1_in] + z*h[turb_split1]
T[pump2_in] = temperature('Steam',P=P[pump2_in], h=h[pump2_in])
v[pump2_in] = volume('Steam',P=P[pump2_in], h=h[pump2_in])
```

The pump outlet conditions are calculated with a known inlet condition and the specified pump efficiency. Again, the quality of the steam leaving the high-pressure feed-water heater system is assumed to be saturated liquid. Similar calculations are done for the high-pressure pump, and the guess value for the mass flow fraction through the feed-water system is removed.

```
"Pump2 outlet: fw2_in"
P[fw2_in] = P[turb_split2]
W_p2 = v[pump2_in]*(P[pump2_in]-P[fw2_in])/eta_pump2
h[fw2_in] = h[pump2_in]-W_p2
W_dot_p2 = W_p2*m_dot_steam*(1-y)
T[fw2_in] = temperature('Steam',P=P[fw2_in],h=h[fw2_in])

"Feedwater 2 outlet: pump3_in"
P[pump3_in] = P[turb_split2]
x[pump3_in] = 0 [-]
h[pump3_in] = enthalpy('Steam',P=P[pump3_in],x=x[pump3_in])
h[pump3_in] = h[fw2_in]*(1-y)+h[turb_split2]*y
T[pump3_in] = temperature('Steam',P=P[pump3_in],h=h[pump3_in])
v[pump3_in] = volume('Steam',P=P[pump3_in], h=h[pump3_in])

"Pump 3 outlet: ph_in"
P[ph_in] = P[turb_in]
W_p3 = v[pump3_in]*(P[pump3_in]-P[ph_in])/eta_pump3
h[ph_in] = h[pump3_in]-W_p3
W_dot_p3 = W_p3*m_dot_steam
T[ph_in] = temperature('Steam',P=P[ph_in],h=h[ph_in])
```

Relatively little information is known about the state of the pre-heater streams, though the mass flow rate of the heat transfer fluid and the inlet temperature of the steam are known. To complete the pre-heater calculations, several parameters have been temporarily assumed, the first of which are the pre-heater HTF inlet temperature and the HTF cold outlet temperature.

```
"Preheater outlet: boil_in"
P[boil_in] = P[turb_in]
c_htf_ph = c_(HTF$,T=(T_htf_ph_in+T_htf_cold)/2)
c_dot_htf_ph = c_htf_ph*m_dot_htf
```

The steam-side calculations require an additional guess value for the pre-heater steam outlet temperature. Both the minimum and maximum pre-heater capacitance rates can then be determined, and the capacitance ratio calculated.

```
c_w_ph = specheat('Steam',T=(T[boil_in]+T[ph_in])/2,P=P[boil_in])
c_dot_w_ph = c_w_ph*m_dot_steam
c_dot_min_ph = MIN(c_dot_htf_ph,c_dot_w_ph)
c_dot_max_ph = MAX(c_dot_htf_ph,c_dot_w_ph)
C_R_ph = c_dot_min_ph/c_dot_max_ph
```

The pre-heater heat-exchanger is modeled in the counter-current configuration, according to the relationship given in Eq. (4.14), which is a modified form of Eq. (4.8) above.

$$\varepsilon_{PH} = \frac{1 - e^{\left[-NTU_{PH}\left(1-C_{R,PH}\right)\right]}}{1 - C_{R,PH} \cdot e^{\left[-NTU_{PH}\left(1-C_{R,PH}\right)\right]}} \tag{4.14}$$

Now that the flow rates through the heat exchanger may deviate from design conditions, the conductance must be adjusted to compensate. This is implemented with Eq.(4.13), previously shown.

```
UA_ph = UA_ph_ref*(m_dot_htf/m_dot_htf_ref)^(0.8)       "The adjusted preheater UA"
NTU_ph = UA_PH/C_dot_min_ph
epsilon_ph = HX('counterflow',NTU_ph,C_dot_min_ph, C_dot_max_ph, 'epsilon')
```

With the effectiveness, the outlet salt temperature, outlet steam enthalpy, and outlet steam temperature can be determined, and the guessed cold heat transfer fluid temperature and pre-heater outlet temperature are replaced with calculated values.

```
"Calculate the heat exchanged"
q_dot_ph = epsilon_ph*C_dot_min_ph*(T_htf_ph_in - T[ph_in])
    q_ph = q_dot_ph/m_dot_steam
"Calculate the cold htf outlet"
T_htf_cold = T_htf_ph_in - q_dot_ph/c_dot_htf_ph
"Calculate the hot steam outlet enthalpy"
h[boil_in] = h[ph_in] + q_dot_ph/m_dot_steam
"Calculate the corresponding outlet steam temperature"
T[boil_in]= temperature('Steam',P=P[boil_in],h=h[boil_in])
```

As the steam enters the boiler under design conditions, it exists as saturated liquid. However, the steam may not be exactly saturated during operation, since at lower heat source flow rates or temperatures the pre-heater conductance might be undersized and fail to conduct enough heat from the hot stream to the cold stream to bring the steam to the saturated condition. Likewise, the pre-heater conductance will be oversized at high-flow or high-temperature salt conditions.

The outlet conditions of the boiler are also known, since the fluid cannot physically leave the boiler until it has reached saturated steam conditions. Again, the temperature of the salt at the boiler inlet is unknown, so a guess value is used to calculate the result.

```
"Boiler outlet: sh_in"
P[sh_in] = P[turb_in]
x[sh_in] = 1 [-]
h[sh_in] = enthalpy('Steam',P=P[sh_in],x=x[sh_in])
c_htf_boil = c_(HTF$,T=(T_htf_boil_in+T_htf_ph_in)/2)
c_dot_htf_boil = c_htf_boil*m_dot_htf
UA_boil = UA_boil_ref*(m_dot_htf/m_dot_htf_ref)^(.8)      "The adjusted preheater UA"
NTU_boil =UA_boil/(m_dot_htf*c_htf_boil)                  "The NTU calculated from known UA and
HTF properties"
epsilon_boil = 1-exp(-NTU_boil)                           "The corresponding effectiveness"
q_dot_boil = epsilon_boil*c_dot_htf_boil*(T_htf_boil_in-T[sh_in])
```

The list of unresolved guess values – including the steam mass flow rate, the heat transfer fluid pre-heater inlet temperature, the heat transfer fluid boiler inlet temperature, and the super-heater steam inlet temperature – is partially resolved as the heat transfer fluid pre-

heater inlet temperature and the steam mass flow rate are calculated. The corresponding guess values are removed from the model.

```
"Calculate the boiler htf outlet temp"
T_htf_ph_in = T_htf_boil_in - q_dot_boil/c_dot_htf_boil
"Calculate the steam outlet enthalpy"
m_dot_steam=q_dot_boil/(h[sh_in] - h[boil_in])
"Find the super-heater steam inlet temp"
T[sh_in] = temperature('Steam',P=P[sh_in],h=h[sh_in])
```

The state of the steam entering the super-heater is the same as that leaving the boiler, so the known *UA* and the minimum capacitance rate can be used to calculate the *NTU* and the effectiveness. Finally, the super-heater heat flow is calculated with the known effectiveness and inlet temperatures. A guess for the high-pressure turbine inlet steam temperature is made on behalf of the required property calculations.

```
"Superheater calculations"
c_htf_sh = c_(HTF$,T=(T_htf_hot+T_htf_boil_in)/2)
c_dot_htf_sh = c_htf_sh*m_dot_htf
c_w_sh = specheat('Steam',P=P[turb_in],T=(T[turb_in]+T[sh_in])/2)
c_dot_w_sh = c_w_sh*m_dot_steam
c_dot_min_sh = MIN(c_dot_htf_sh,c_dot_w_sh)
c_dot_max_sh = MAX(c_dot_htf_sh,c_dot_w_sh)
C_R_sh = c_dot_min_sh/c_dot_max_sh
UA_sh = UA_sh_ref*(m_dot_htf/m_dot_htf_ref)^(.8)
NTU_sh = UA_sh/c_dot_min_sh
epsilon_sh = HX('counterflow',NTU_sh,c_dot_min_sh,c_dot_max_sh,'epsilon')
"heat exchanged"
q_dot_sh = epsilon_sh*c_dot_min_sh*(T_htf_hot-T[sh_in])
    q_sh = q_dot_sh/m_dot_steam
```

Two guess temperatures remain unresolved: the boiler salt inlet temperature and the turbine steam inlet temperature. Both can be determined with energy balances on the super-heater heat exchanger. The HTF boiler inlet temperature can be determined by calculating the heat flow into the HTF fluid, while the turbine steam inlet temperature can be found by calculating the change in enthalpy associated with that same heat flow.

```
"htf outlet"
T_htf_boil_in = T_htf_hot-q_dot_sh/c_dot_htf_sh
"steam"
h[turb_in]=h[sh_in]+q_sh                                    "negates T[turb_in] guess value"
```

The cycle is now completely defined and represents the Rankine cycle as it operates under varying conditions. Final cycle performance characteristics such as total power output and cycle thermal efficiency are calculated.

```
"cycle summary"
q_dot_hot_tot = q_dot_ph+q_dot_boil+q_dot_sh
W_dot_cycle = W_dot_tot+W_dot_p1+W_dot_p2+W_dot_p3
eta_thermal = W_dot_cycle/Q_dot_hot_tot
```

A comparison of the performance model output to the design model output shows excellent agreement. Differences in the values are due chiefly to the specification of the boiler and condenser outlet conditions as saturated steam and saturated liquid, respectively. These states deviate slightly from the design case, since the specified design steam conditions are either slightly sub-cooled (in the case of the condenser) or slightly super-heated (in the case of the boiler). Table 15 shows several selected cycle parameters and outputs from the performance cycle as compared to the equivalent in the design cycle model.

Table 15: Model output comparison for the Rankine cycle performance model and the Rankine cycle design model

DESCRIPTION	VARIABLE	PERFORMANCE MODEL VALUE	DESIGN MODEL VALUE
HTF hot inlet temperature	$T_{htf,hot}$	838.7 [K]	838.7 [K]
HTF boiler inlet temperature	$T_{htf,boil,in}$	765.9 [K]	765.8 [K]
HTF pre-heater inlet temperature	$T_{htf,ph,in}$	623.7 [K]	622.1 [K]
HTF cold outlet temperature	$T_{htf,cold}$	571.5 [K]	560.9 [K]
Steam mass flow rate	$\dot{m}_{steam}$	12.27 [kg/s]	11.84 [kg/s]
Cycle electric power output	$\dot{W}_{cycle}$	10.0 [MW]	10.0 [MW]
Cycle thermal efficiency	η_{cycle}	0.3352	0.3316
Condenser cooling water outlet temp.	$T_{cw,out}$	311.6 [K]	311.8 [K]

To better characterize the performance model output and to project the behavior of the 10*MWe* plant to other plant sizes, the output values in Table 15 are non-dimensionalized according to their design reference values.

Cycle power output, cycle heat addition, and thermal efficiency can be scaled directly with their design values. The cycle heat addition is defined as the rate of heat flow from the heat transfer fluid through the heat exchangers to the steam in the Rankine cycle.

$$\dot{W}_{ND} = \frac{\dot{W}_{cycle}}{\dot{W}_{design}} \tag{4.15}$$

$$\dot{q}_{heat,ND} = \frac{\dot{q}_{heat}}{\dot{q}_{heat,ref}} \tag{4.16}$$

$$\eta_{thermal,ND} = \frac{\eta_{thermal}}{\eta_{design}} \tag{4.17}$$

The temperature difference that drives the steam mass flow rate in the Rankine cycle is the difference between the hot heat transfer fluid inlet temperature and the saturation temperature of the steam boiler pressure. Therefore, HTF temperatures are non-dimensionalized according to this temperature difference. For the non-dimensional cold heat transfer fluid outlet temperature, the expected value near design conditions is minus one.

$$T_{htf,hot,ND} = \frac{T_{htf,hot} - T_{sat,ref}}{T_{htf,hot,ref} - T_{sat,ref}} \tag{4.18}$$

$$T_{htf,cold,ND} = \frac{T_{htf,cold} - T_{sat,ref}}{T_{sat,ref} - T_{htf,cold,ref}} \tag{4.19}$$

The cycle mass flow rates – including the steam mass flow rate, the heat transfer fluid mass flow rate, and the cooling water mass flow rate – are non-dimensionalized according to their nominal rates at design conditions.

$$\dot{m}_{steam,ND} = \frac{\dot{m}_{steam}}{\dot{m}_{steam,ref}} \tag{4.20}$$

$$\dot{m}_{cw,ND} = \frac{\dot{m}_{cw}}{\dot{m}_{cw,ref}} \tag{4.21}$$

$$\dot{m}_{htf,ND} = \frac{\dot{m}_{htf}}{\dot{m}_{htf,ref}} \tag{4.22}$$

Finally, the cooling water outlet temperature is non-dimensionalized according to the nominal temperature drop across the cooling water at design conditions.

$$\begin{aligned} T_{cw,ND} &= \frac{T_{cw,out} - T_{cw,in}}{\Delta T_{cw,ref}} \\ where: & \\ \Delta T_{cw,ref} &= T_{cw,out,design} - T_{cw,in,design} \end{aligned} \tag{4.23}$$

4.3 The Rankine Cycle Regression Model

The performance of solar energy systems is necessarily dependent upon the weather. Long-term analyses of solar power generation systems require repeated calculations at short time intervals over extended periods of time. Consequently, the computational effort required to simulate a solar system power generation is a concern. The most computationally intensive component in a solar power tower electrical generation system is the Rankine cycle and its associated heat exchangers. The model that determines the amount of electrical power that can be produced given the mass flow rates and temperatures of the hot (heat transfer fluid) and cold (cooling water) fluids involves many implicit relationships that require iterative calculations to solve. These iterative calculations are done automatically in EES, but require significant coding in order for them to be solved in TRNSYS with Fortran. An additional but perhaps minor consideration is the steam property database. Steam properties are available in TRNSYS but the formulation used to provide these properties is empirical rather than equation of state-based, as in EES.

Discussion in the previous section identified a number of non-dimensional parameters that can be used to characterize the behavior of a Rankine cycle over a range of operating conditions. These non-dimensional parameters are used here to develop a polynomial

regression model that represents the behavior of the Rankine cycle calculated with a detailed model in EES for a wide range of operating conditions.

The goal of using non-dimensional parameters in the regression model is to obtain a model that can be applied to any Rankine power cycle over a wide range of operating conditions. The model requires, as input, several parameters related to their specific Rankine cycle configuration, including:

- Rankine plant design electric power output
- Rankine cycle efficiency at design conditions
- Plant inlet and outlet heat transfer fluid temperatures at design conditions
- Heat transfer fluid mass flow rate

These values should be readily available to the plant designer through the power cycle manufacturer as they describe the system performance at its reference or design condition. The regression model will then calculate the change in the output variables (e.g., power, heat rejection, return salt temperature) with respect to the deviation of the inputs from their reference design condition values.

The regression model was constructed by correlating the non-dimensionalized outputs from a detailed Rankine cycle model developed in EES. These outputs of interest include:

- Actual cycle electric power output
- Cycle thermal efficiency
- Heat transfer fluid outlet temperature
- Condenser cooling water outlet temperature
- Cycle heat addition rate

These outputs define the cycle's interaction with the surrounding components, and are a function of the conditions under which the cycle is operating. The following sections describe the regression technique and the behavior of the model under variable

conditions, including a validation of the model with respect to other Rankine cycle configurations.

4.3.1 Power cycle regression methodology

To obtain an accurate multiple-variable regression model of a system, first-order effects of the main input variables as well as higher order interaction effects between the input variables need to be considered. However, the Rankine cycle is a complex system and analysis shows that not all main effects or interaction effects have significant impact on the outputs. For example, though the condenser cooling water flow rate will have a significant impact on the cooling water outlet temperature, it may have little or no impact on the heat-transfer fluid cycle outlet temperature. Depending on the thermal size of the reference condenser, the cooling water flow rate may have little or no effect on the power output as well. Inclusion of non-significant effects in a regression model needlessly increases model complexity without providing additional accuracy. Therefore, a statistical analysis is conducted to sift out the significant effects while demonstrating the relative dependence of the various outputs on the model parameters.

A four-factor full-factorial experiment at two levels is used, with the four input factors being the heat transfer fluid cycle hot inlet temperature, the condenser cooling water inlet temperature, the cooling water mass flow rate, and the heat transfer fluid mass flow rate. These are the inputs that vary during the operation of solar-driven power system and affect the performance of a Rankine cycle. The output variables are the previously mentioned cycle electric power output, the cycle thermal efficiency, the heat transfer fluid outlet temperature, and the condenser cooling water outlet temperature. Additionally, the cycle heat input rate is calculated and considered as an output since it can be used indirectly to determine other cycle performance factors. The full-factorial experiment tests the four input variables at their high and low values, and at every possible variable level combination. At two levels and four factors, a full factorial analysis requires 2^4, or sixteen runs. The experimental design table is shown in Table 16.

Table 16: Rankine cycle regression analysis factorial experiment design table. The (+) value indicates the higher of the two levels, (-) indicates the lower value.

	A	B	C	D
	$T_{htf,hot}$	$T_{cw,in}$	$\dot{m}_{cw}$	$\dot{m}_{htf}$
1	-	-	-	-
2	-	-	-	+
3	-	-	+	-
4	-	-	+	+
5	-	+	-	-
6	-	+	-	+
7	-	+	+	-
8	-	+	+	+
9	+	-	-	-
10	+	-	-	+
11	+	-	+	-
12	+	-	+	+
13	+	+	-	-
14	+	+	-	+
15	+	+	+	-
16	+	+	+	+

Selection of the high and low variable values requires specific knowledge of the cycle operating conditions, and care must be taken to choose values that adequately reflect the reasonable upper and lower limits of cycle operating conditions. For the cycle used in this analysis, the upper and lower limits are presented in Table 17.

Table 17: Factorial experiment upper and lower values.

ITEM	VARIABLE	UNITS	DESIGN VALUE	(-) ND	(+) ND	(-)	(+)
A	$T_{htf,hot}$	K	838.7	0.6	1.1	736.9	864.2
B	$T_{cw,in}$	C	27.6	-	-	18	45
C	$\dot{m}_{cw}$	kg/s	438.1	0.85	1.1	372.4	481.9
D	$\dot{m}_{htf}$	kg/s	81.69	0.5	1.5	36.7	110

The performance model is run according to the information in Table 16 and Table 17, and the output is analyzed to determine the significance of each effect. The main effects of the input variables are calculated by taking the difference between the averaged upper values and the averaged lower values. For example, the main effect for the heat transfer fluid hot inlet temperature (A) is calculated with the following relationship.

$$ME\left(A\right)_j = \left.\frac{\sum_{1}^{n^+} Y_j^{A+}}{n^+} - \frac{\sum_{1}^{n^-} Y_j^{A-}}{n^-}\right\} \; for \;\; j = 1..N_Y \tag{4.24}$$

where: Y_j represents the response variable being measured

N_Y is the number of response variables

n^+ is the number of runs for variable Y at the upper level

n^- is the number of runs for variable Y at the lower level

For each measured response, four main effects are calculated: A, B, C, and D, representing heat transfer fluid cycle hot inlet temperature, the condenser cooling water inlet temperature, the cooling water mass flow rate, and the heat transfer fluid mass flow rate, respectively. Table 18 shows the calculated main effects for each variable. Note that the main effects have been calculated using the non-dimensionalized version of the parameters.

Table 18: Factorial experiment calculated Main Effects

	A	**B**	**C**	**D**
	$T_{htf,hot}$	$T_{cw,in}$	$\dot{m}_{cw}$	$\dot{m}_{htf}$
$\dot{W}_{cycle}$	0.48261	0.09214	0.00855	0.78479
$\dot{q}_{heat}$	0.48076	0.01089	0.00096	0.80932
$T_{htf,cold}$	0.10946	0.23816	0.01624	1.00076
η_{therm}	0.03923	0.09566	0.00650	0.01557
$T_{cw,out}$	0.50066	0.03166	0.23293	0.85731

The interaction effects between input variables can be calculated in a similar fashion to the main effects, except that the values are sorted not only by the upper and lower levels of a single variable, but also by the upper and lower values of both variables involved in the interaction. This relationship is illustrated in the following equation showing the interaction calculation for effects *A* and *B*.

$$INT\left(AB\right)_j = \frac{1}{2}\left\{\left(\frac{\sum_{1}^{n^+/2} Y_j^{A+|B+}}{n^+/2} - \frac{\sum_{1}^{n^-/2} Y_j^{A+|B-}}{n^-/2}\right) - \left(\frac{\sum_{1}^{n^-/2} Y_j^{A-|B+}}{n^-/2} - \frac{\sum_{1}^{n^+/2} Y_j^{A-|B-}}{n^+/2}\right)\right\} \quad for \quad j = 1..N_Y \quad (4.25)$$

These two-variable interactions are calculated for all of the possible effect pairs, including *AB*, *AC*, *AD*, *BC*, *BD*, and *CD*. Three and four variable interaction effects, while possible to compute, are unlikely to be statistically significant. This analysis will reflect that fact, and so these relationships will not be further discussed. The variable interaction effects are calculated in Table 19.

Table 19: Two, three and four-variable interaction effects for Rankine statistical analysis.

	TWO VARIABLE INTERACTIONS					
	AB	**AC**	**AD**	**BC**	**BD**	**CD**
$\dot{W}_{cycle}$	0.01222	0.00234	0.10802	0.00001	0.02278	0.00352
$\dot{q}_{heat}$	0.00080	0.00022	0.11601	0.00001	0.00252	0.00039
$T_{htf,cold}$	0.01779	0.00363	0.11864	0.00032	0.00579	0.00424
η_{therm}	0.00016	0.00104	0.01512	0.00002	0.00127	0.00185
$T_{cw,out}$	0.00521	0.03296	0.12529	0.00201	0.00809	0.05618

	THREE VARIABLE INTERACTIONS				FOUR VARIABLE...
	ABC	**ABD**	**ACD**	**BCD**	**ABCD**
$\dot{W}_{cycle}$	0.00000	0.00583	0.00194	0.00001	0.00000
$\dot{q}_{heat}$	0.00001	0.00036	0.00018	0.00001	0.00000
$T_{htf,cold}$	0.00013	0.00133	0.00201	0.00006	0.00001
η_{therm}	0.00000	0.00033	0.00062	0.00002	0.00000
$T_{cw,out}$	0.00066	0.00253	0.01679	0.00102	0.00032

The two-variable interactions are more clearly illustrated through interaction plots, where both of the variable effects can be mapped. Figure 35 shows the interaction plots for the Cycle Power Output response.

Since the total effect of the variables on each response is additive, the two lines will be parallel in the case that the interaction between the variables is negligible. In other words, for two theoretical variables *X* and *Y* with no interaction, the relative effect of changing variable *X* from its high to low level is the same at the low level of *Y* as it is at the high level of *Y*. This result is well illustrated in the case of *Plot BC* in Figure 35, where cycle power output is measured in response to change in two variables while the other variables are held constant. For variables where interaction is not negligible, the interaction effect is manifested in the plot through non-parallelism, as in the case of *Plot AD*.

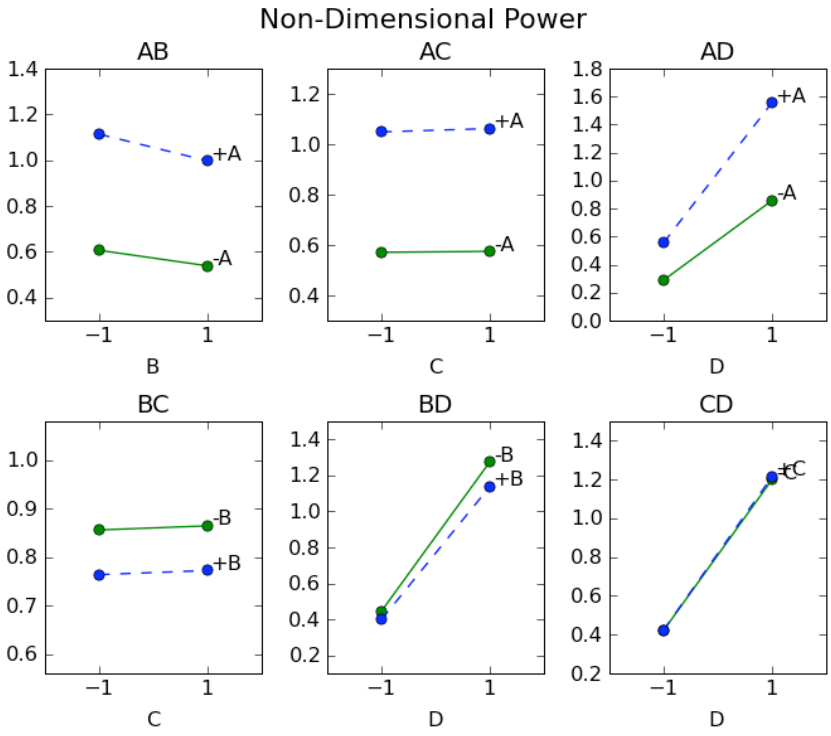


Figure 35: Interaction plots for the Non-Dimensional Power response variable (y-axis). The variables that are changing are represented by: A=HTF hot inlet temp, B=CW inlet temp, C=CW Mass flow, D=HTF mass flow. Other variables are held constant.

The interaction plots for the remaining three response variables are shown in Appendix C. All of the responses have been non-dimensionalized.

While the qualitative analysis provided by the interaction plots is useful in determining variable interactions, a thorough quantitative analysis is also required to proceed in the regression modeling with confidence. This analysis is performed using Lenth's method, which compares the calculated effects to the pseudo-standard error (PSE) – an estimator of the experimental standard deviation in the case where each run is un-replicated. This value is calculated using the following relationship (Wu and Hamada, 2000):

$$PSE = 1.5 \cdot median_{\{\theta_i | < 2.5 s_0\}} |\theta_i| \tag{4.26}$$

The median is computed among the $|\theta_i|$ with $|\theta_i|$ $2.5s_0$, and $s_0 = 1.5 \cdot median|\theta_i|$, and $|\theta_i|$ represents the absolute value of each effect – one, two, three and four-variable effects included. In other words, these values indicate the dependence of a measured response variable on an input variable, and are calculated as shown above in Equations (4.24) and (4.25). The test-statistics can be obtained by dividing each effect by the *PSE*, as follows.

$$t_{PSE,i} = \frac{\theta_i}{PSE} \tag{4.27}$$

The statistical significance of each of the effects is compared to the cutoff value at the desired level of significance. A significance of 95% certainty was chosen for this experiment, corresponding to a cutoff value of 2.16. A higher level of certainty was not selected because in this analysis, it is better to erroneously include an effect that is not actually significant than it would be to erroneously exclude an effect that is – in fact – significant. A test statistic that is greater than this cutoff value indicates significance for that effect, with a 5% chance that the statistic has erroneously been deemed significant. The calculated *PSE* values and test statistics are presented in Table 20, with the significant values emphasized.

Table 20: The regression analysis significance test-statistics are shown. Significant values above the 95% cutoff value of 2.16 are emphasized.

	$\dot{W}_{cycle}$	$\dot{q}_{heat}$	$T_{htf,cold}$	η_{therm}	$T_{cw,out}$
PSE	*0.003203*	*0.000298*	*0.001997*	*0.000363*	*0.007813*
A	150.676	1613.611	54.817	108.153	64.077
B	28.767	36.568	119.268	263.722	4.053
C	2.670	3.228	8.131	17.906	29.812
D	245.021	2716.423	501.164	42.914	109.723
AB	3.814	2.690	8.911	0.436	0.667
AC	0.729	0.733	1.818	2.866	4.219
AD	33.724	389.370	59.415	41.682	16.035
BC	0.003	0.037	0.160	0.059	0.257
BD	7.111	8.463	2.901	3.497	1.036
CD	1.099	1.307	2.125	5.097	7.190

With each factor being examined only at a high and low level, the regression model for each effect must be assumed linear unless additional data points are introduced. The factorial experimental model assumes a linear relationship between the response variables and varying effect values. Although linear behavior may not be the case over the entire range of applicability, the behavior can be modeled as linear sufficiently well such that the factorial experiment analysis still detects dependence of a response on an effect.

To obtain a more accurate regression model, each significant main and interaction effect is fit to a data set that consists of many effect levels, instead of the two used in the original analysis. This fit no longer needs to be linear, and so it is fit to the most appropriate regression curve type. The regression curves for the heat addition and cycle power output as a function of varying heat transfer fluid inlet temperature are included as an example in Figure 36.

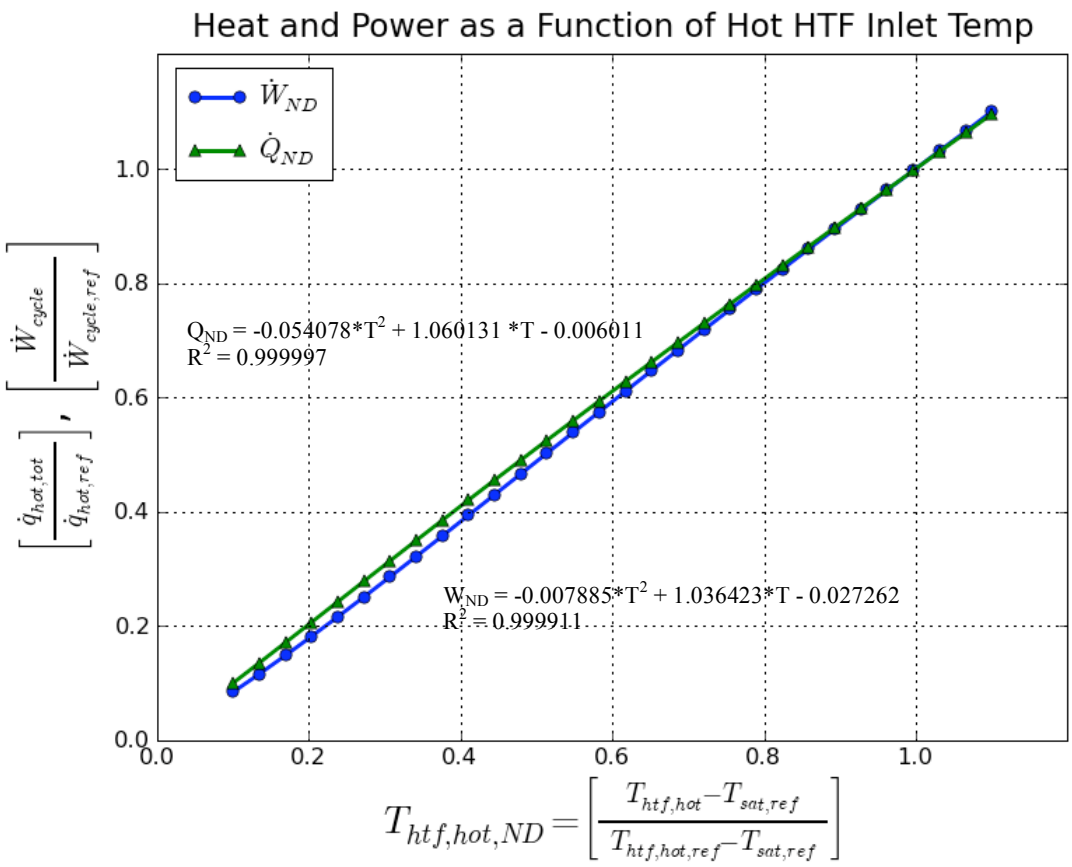


Figure 36: Regression of the main effects involving variation of the heat transfer fluid inlet temperature. The effect of variation in the heat-transfer fluid inlet temperature is shown for only the cycle power and heat addition response variables. In the equations, y is the non-dimensional heat or power, and x is the non-dimensional inlet temperature.

In similar fashion to the regression in Figure 36, the other effects are correlated with the response variables to determine how the response variables depend on variation in operating conditions. The regression modeling of the interaction effects is somewhat less straightforward than the main effect modeling. To help illustrate the interaction effect methodology, Figure 37 is included below.

Each interaction effect is a measure of the interdependence of two or more main effects. The measured response variable, which is modeled to be a function of several additive main effect variables, may lose accuracy when the magnitude of the additive portion of one main effect depends on the value of another main effect. This problem can be

remedied through the inclusion of the interaction effect as illustrated in Figure 37 where two hypothetical main effects – θ_1 and θ_2 – are presented on an interaction plot.

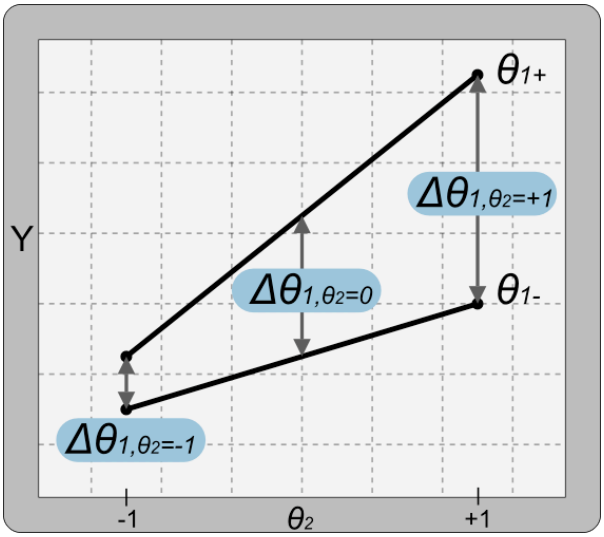


Figure 37: Interaction effect regression methodology. The interaction is modeled by considering how the difference between the responses varies as a function of effect level.

The magnitude of the effect of θ_1 on the measured response variable when other variables are at their nominal levels ($\theta_2 = 0$) is considered to be the "main effect" of θ_1. This is shown as the difference between θ_{1+} and θ_{1-}, or $\Delta\theta_{\theta_2=0}$. This difference is calculated at a number of points between the limits of $\theta_2 = -1$ and $\theta_2 = +1$, and the difference is then expressed in terms of its magnitude relative to the original difference at $\theta_2 = 0$.

In other words, a correlation is developed for which the difference between the values of θ_{1+} and θ_{1-} at various θ_2 is expressed as a fraction of the main effect $\Delta\theta_{\theta_2=0}$. This factor is intended to be used to scale the main effect involved in the interaction. This is shown in Eq. (4.28) for two effects, θ_j and θ_k.

$$\theta_j = ME\left(\theta_j\right) \cdot \frac{\Delta\theta_{j,\theta_k}}{\Delta\theta_{j,\theta_k=0}} \tag{4.28}$$

Figure 38 shows the relationship between cycle heat addition and heat transfer fluid flow rate at three values of heat transfer fluid inlet temperature. The subplot in Figure 38

indicates the distance between any two of the plotted lines at a given heat transfer fluid flow rate, as scaled by the distance between those same lines at a non-dimensional flow rate of one. The correlation in the subplot gives the value of $\frac{\Delta\theta_{j,\theta_k}}{\Delta\theta_{j,\theta_k=0}}$ in Eq. (4.28).

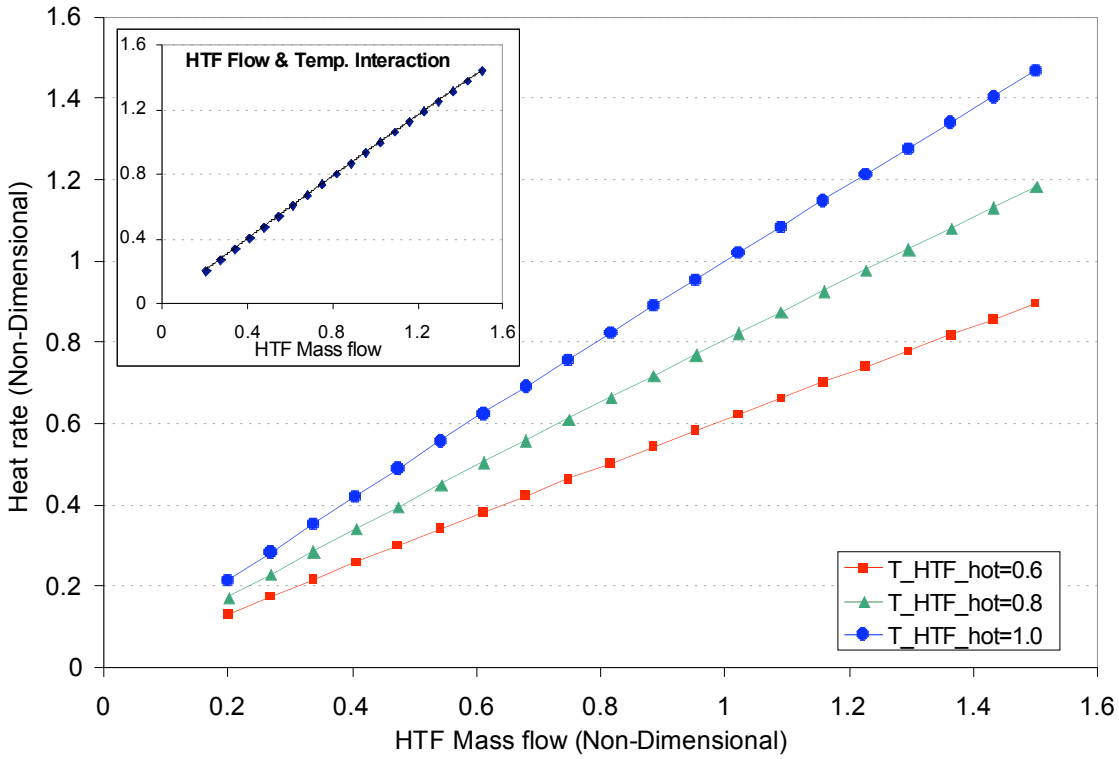


Figure 38: The interaction between the heat transfer fluid (HTF) inlet temperature and the heat transfer fluid flow rate, as it affects the cycle heat addition (main plot). The interaction between the flow rate and inlet temperature is plotted (upper left).

An interaction effect regression analysis similar to this example is performed for each of the significant interaction effects, as listed in Table 20. Finally, the main effects, which have been modified to accommodate any interaction, are added together (minus the value about which they are normalized) and the end result is a factor that indicates the non-dimensional value of the response variable.

The results of this analysis provide insight into the behavior of the cycle over the range of operation. While some of the response variables behave linearly or approximately linearly with consistent second order interaction effects, others do not. One case where

the response variable behavior is highly non-linear is cycle thermal efficiency. Figure 39 shows this efficiency as a function of heat transfer fluid mass flow rate at three separate heat transfer fluid inlet temperatures.

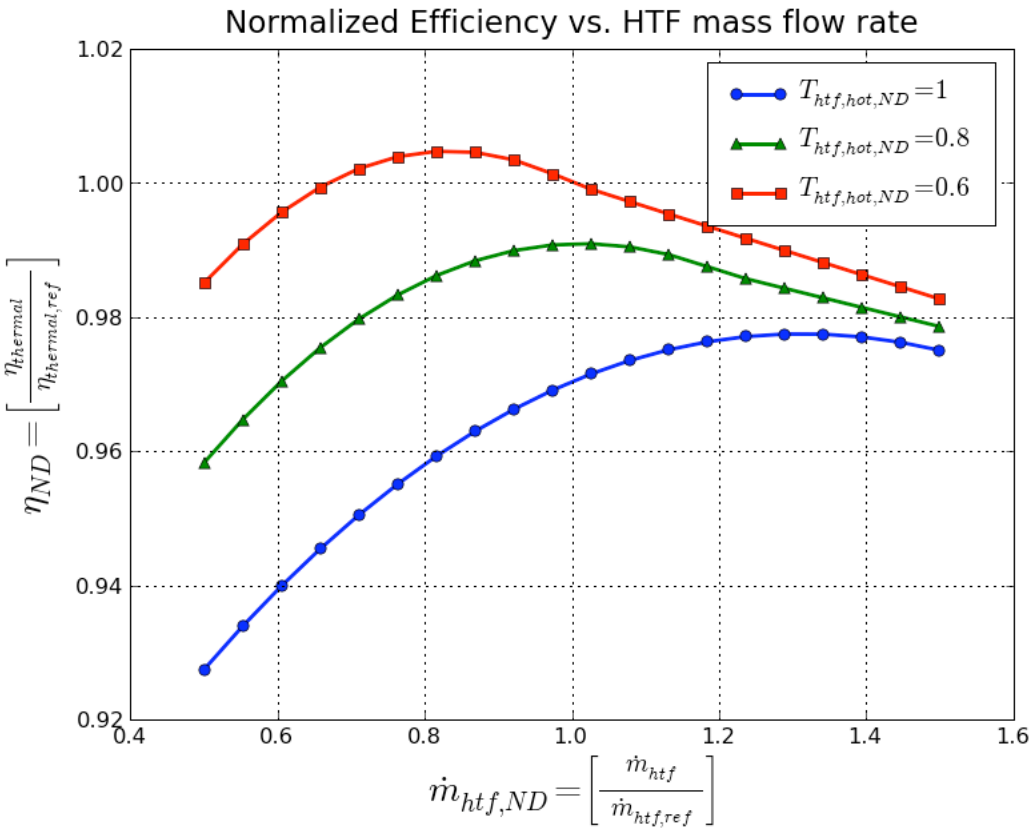


Figure 39: Normalized thermal efficiency of the Rankine cycle as a function of HTF mass flow rate at three HTF inlet temperatures. The normalized efficiency is simply the cycle efficiency at operating conditions scaled by the efficiency at design conditions.

This behavior reflects the opposing effects of degrading turbine efficiency and improving heat exchanger effectiveness as the mass flow rate of steam through the cycle varies. Since the mass flow rate of steam is controlled with an energy balance in the boiler such that it leaves at a quality of one, an increasing heat transfer fluid mass flow rate and corresponding increase in the heat transfer rate in the boiler will increase the steam mass flow rate. However, this increased steam mass flow rate reduces the number of transfer units provided by the heat exchanger, and the thermal performance suffers. This concept is shown by examining the relationship between the physical size of the heat-exchanger,

the capacitance rate ($\dot{c}_{\min} = c_p \dot{m}_{fluid}$), and the number or transfer units (*NTU,* the non-dimensional size of the heat exchanger).

$$NTU = \frac{UA}{\dot{C}_{\min}} \tag{4.29}$$

It is clear that as the mass flow rate of the fluid through the heat exchanger increases, the capacitance rate increases. The *UA* of the heat exchanger also varies slightly with mass flow rate, but not significantly enough to match the variation in the capacitance rate. As the mass flow rate increases, the denominator of the fraction in Eq. (4.29) increases, causing a reduction in the non-dimensional size of the heat exchanger.

This complex behavior is very difficult to accurately correlate directly, and a poorly correlated efficiency curve (or any other curve) will lead to non-physical results in which the cycle output does not satisfy an energy balance. A solution to this problem is apparent by noting that both the cycle power output and the heat transfer rate maintain linearity over the range of operation (see Figure 36 and Figure 38). Since the cycle efficiency and all of the other correlated cycle outputs can easily be calculated knowing the power output and heat transfer rate, only these variables need be directly correlated. This process avoids over-specification of the cycle, and maintains an energy balance on the cycle output.

Figure 40 shows the importance of this technique. The result of directly correlating efficiency (which does not work well) is shown on the left, while the method of correlating the heat transfer rate to the cycle and the output power and then back-calculating efficiency is shown on the right.

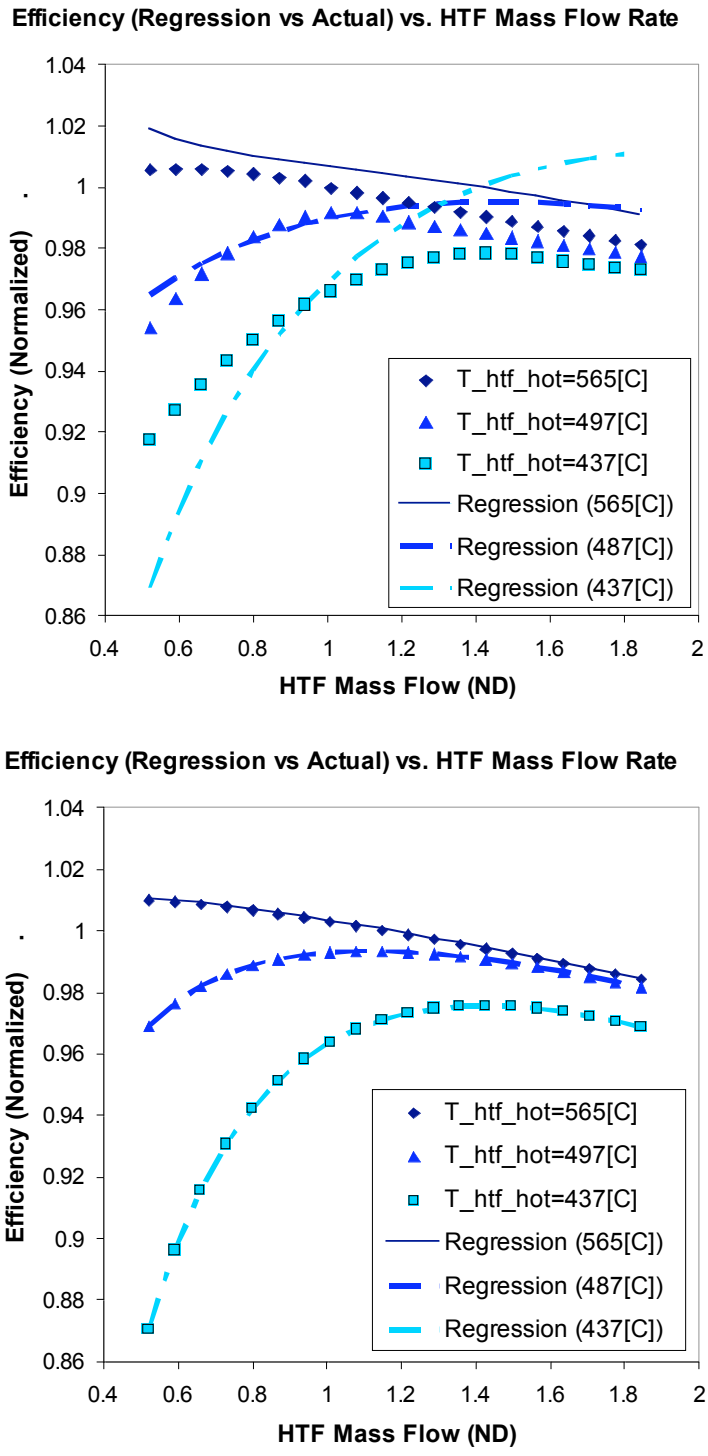


Figure 40: Demonstration of regression methodology. The results of directly correlating efficiency (TOP) vs. correlating heat transfer and work and calculating efficiency (BOTTOM).

4.3.2 Regression Model Verification

The development of the regression model required the assumption of significant and insignificant factors and variable interactions based on statistical analysis. Because certain effects were dismissed as inconsequential to the outcome of the model, verification is required to compare the resulting behavior of the regression model to the actual thermodynamic model in EES. Applicable ranges of operation for the regression model can also be provided by comparing the error between the regression model and the EES model.

Table 21 summarizes the reference design conditions for an analysis that compares the two models over a range of heat transfer fluid mass flow rates and inlet temperatures. The models were tested for three different heat transfer fluid temperatures at along a series of heat transfer fluid mass flow rates.

Table 21: Values used for comparative analysis of the Regression and EES Performance model.

Description	*Parameter*	*Value*	*Units*
Reference cycle power production (net)	$\dot{W}_{design}$	10.00	*MWe*
Reference thermal efficiency	η_{design}	33.522	*%*
Reference hot HTF inlet temperature	$T_{htf,hot,ref}$	565.6	*°C*
Reference cold HTF outlet temperature	$T_{htf,cold,ref}$	298.3	*°C*
Reference condenser temperature rise	$\Delta T_{cw,ref}$	10.82	*°C*
Cycle boiler pressure	P_{boil}	100	*Bar*
HTF specific heat linear regression slope	c_{HTF1}	1.729×10^{-4}	$\frac{J}{kg-K^2}$
HTF specific heat linear regression intercept	c_{HTF2}	1.4472	$\frac{J}{kg-K}$

The results of this analysis show that the regression model matches well with the output of the EES model, with percentage errors in output generally limited to a few percentage points. The modeled power output as a function of HTF mass flow rate at three HTF inlet temperatures is shown in Figure 41. The error associated with the difference between the regression and EES models is also plotted for each HTF temperature. Error is defined as shown in Eq.(4.30).

$$Err = \frac{\dot{W}_{regression} - \dot{W}_{EES}}{\dot{W}_{Design}} \times 100\% \quad (4.30)$$

The magnitude of the error increases as the deviation from design conditions increases. The highest error is observed at low HTF mass flow rates and cooler HTF inlet temperatures.

Besides verifying that the regression and EES models are internally consistent, a validation of the model using external data is helpful in assessing the quality of the regression model. To accomplish this, the Rankine cycle model was compared to limited data provided by the Solar II Rankine cycle, as presented in Pacheco (2002). Reference conditions for this analysis are the design conditions for the Solar II Rankine cycle, specified in Table 22.

Data taken at steady state from the Solar II Rankine cycle are compared to the predicted data using the regression model. Two target salt inlet temperatures were used over a range of target HTF flow rates. The results of the comparison are most clearly presented as the relationship between cycle power output and the HTF flow rate as shown in Figure 42. The results of the comparison show good agreement, with the largest error at low flow rates, as expected.

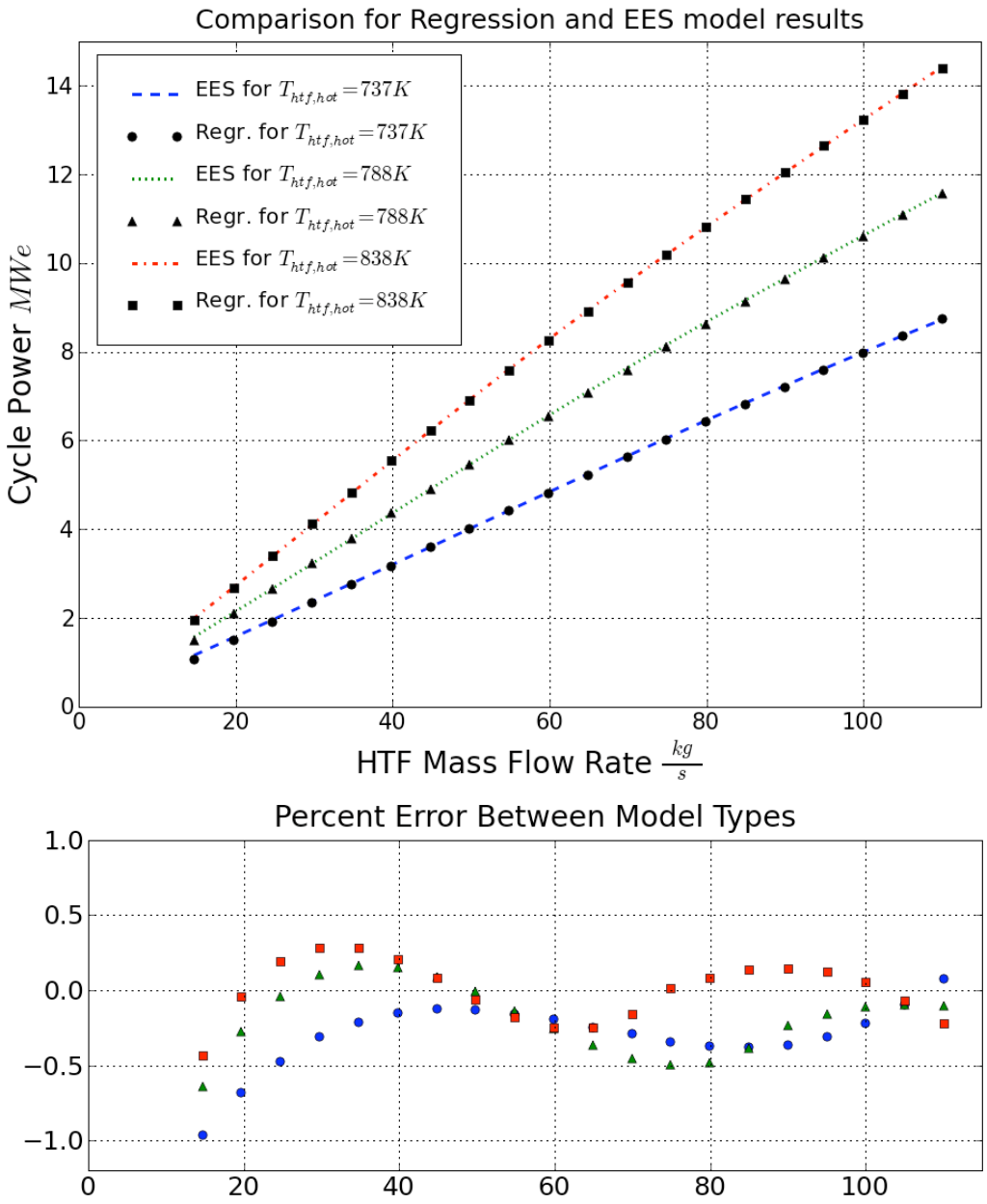


Figure 41: The power output as a function of HTF mass flow rate for three HTF Inlet temperatures. Results of analysis with EES Performance model and Regression model are compared (TOP). The associated error is calculated and shown (BOTTOM).

Table 22: Reference design conditions for the comparison with the Solar II Rankine cycle

Description	*Parameter*	*Value*	*Units*
Reference cycle power production (net)	$\dot{W}_{design}$	12.00	*MWe*
Reference thermal efficiency	η_{design}	33.8	*%*
Reference hot HTF inlet temperature	$T_{htf,hot,ref}$	565.6	*°C*
Reference cold HTF outlet temperature	$T_{htf,cold,ref}$	288	*°C*
Reference condenser temperature rise	$\Delta T_{cw,ref}$	11	*°C*
Cycle boiler pressure	P_{boil}	110	*Bar*
HTF specific heat linear regression slope	c_{HTF1}	1.729×10^{-4}	$\frac{J}{kg-K^2}$
HTF specific heat linear regression intercept	c_{HTF2}	1.3672	$\frac{J}{kg-K}$

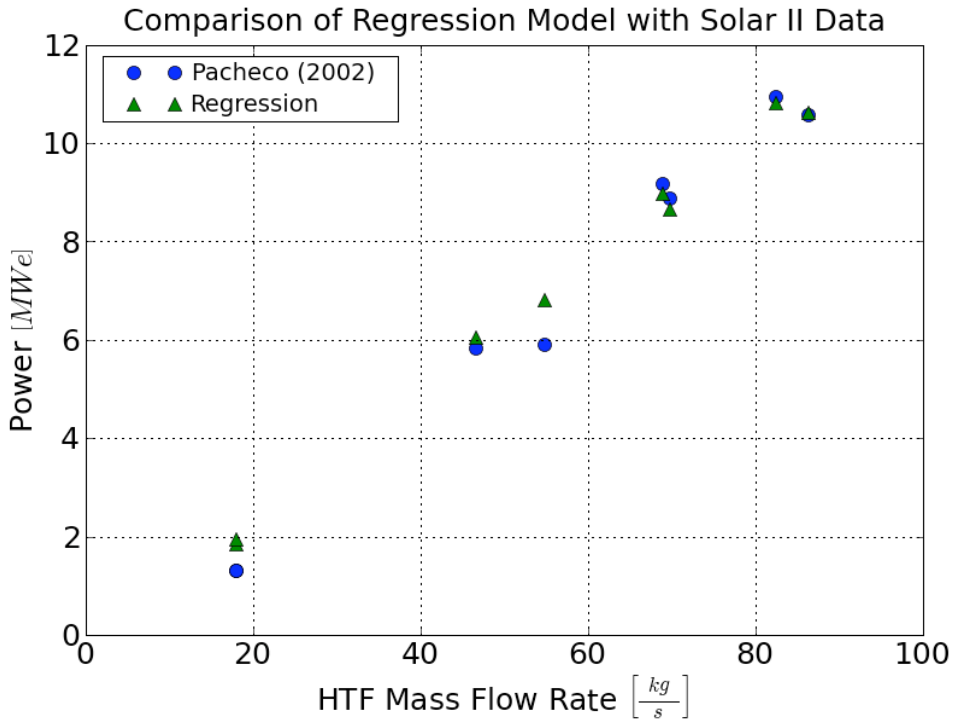


Figure 42: The results of a comparison between Solar II Rankine cycle data (Pacheco, 2002) and the predicted value.

An additional study was conducted to reinforce the regression model using the predictive fundamental model developed by Lippke (1995) for the Solar II cycle. The Lippke model was constructed to provide insight into the Solar II cycle before it went into

operation and had not yet been validated with Solar II operating experiments. However, this investigation is helpful in providing a comparison between two Rankine cycle models. The results of this analysis are shown below in Figure 43.

The regression model also shows good agreement with the results from the Lippke model, with small error in the power output and very similar results for the heat transfer fluid outlet temperature. These analyses some level of assurance that the regression model compares favorably to cycles in operation.

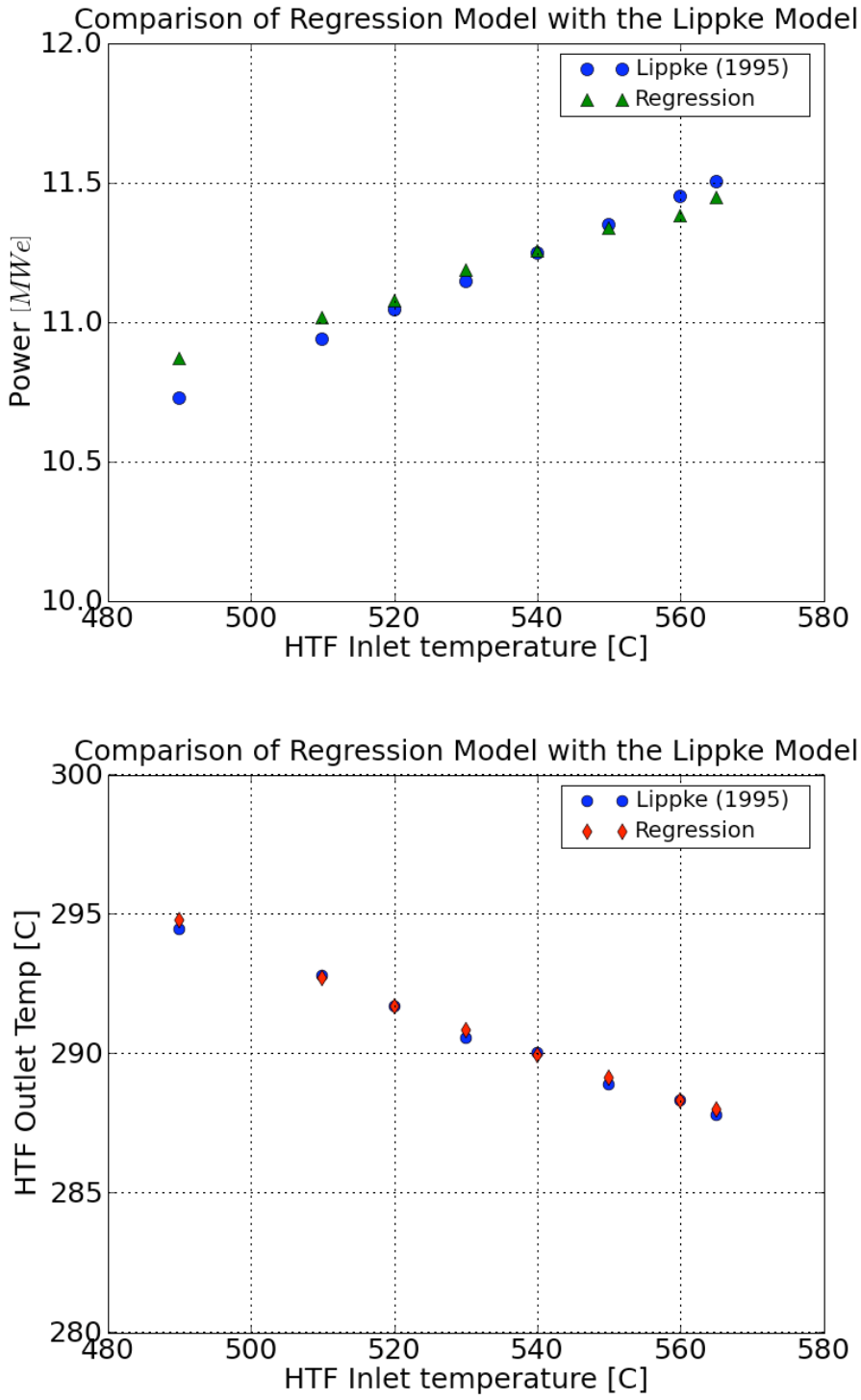


Figure 43: Results of the Lippke model comparison, showing the output power (TOP) and the HTF outlet temperature (BOTTOM) for both models.

5 Plant Sizing and Optimization

A large amount of detailed information is required to accurately evaluate the power tower plant concept, including receiver geometry, heliostat field layout, and heat transfer fluid loop design. In the usual case, detailed plant information is not known a priori – only general sizing requirements and the operating environment are known. An ideal power tower plant model must be able to gather the relevant inputs and simulate plant operation over long-term periods.

The inputs required for simulations of a power tower in TRNSYS include specific receiver geometry, heliostat field net efficiency as a function of solar position, and the flux distribution on the receiver as a function of solar position for the specific plant geometry in question. This information is unique to each plant, and obtaining it requires an analysis of the heliostat field being coupled to the specific receiver geometry. When considering an additional requirement that the plant be optimized with respect to the heliostat field and receiver configurations, the task of developing a modeling tool to accommodate these needs becomes particularly challenging.

A thorough literature review was conducted to assess the capability of existing component or plant models, and several potentially useful models were found. These models include the University of Houston code for heliostat field layout (Lipps and Vant Hull, 1977), HFLCAL (Kiera and Schiel, 1989) for heliostat field layouts, MIRVAL (Leary and Hankins, 1979) for comparison of heliostat field layouts, SolTRACE (Wendelin, 1989) for optical analysis, and DELSOL3 (Kistler 1986) for Central Receiver System (CRS) plant layout.

Because the specific purpose of the DELSOL3 code is to generate an optimized plant design for a central receiver system, it is considered the best match for incorporating into this project. DELSOL3 is capable of generating a specific receiver geometry that is matched to a heliostat field layout. It provides useful output for long-term simulation

such as an array of net heliostat field efficiency as a function of solar position and flux distribution on the receiver surface as a function of solar position. Other advantages of using DELSOL3 include the extensive documentation that accompanies the program, and the fact that the code is written in Fortran, allowing for easy integration with other Fortran-based programs, including TRNSYS.

The following section presents a brief overview of DELSOL3 and discusses some of its main features. To augment the usefulness of this code, a graphical user interface (GUI) was developed to handle some primary tasks in DELSOL3. The final section presents this interface and gives a detailed look at the methodology for integrating DELSOL3 into this research.

5.1 Plant Design with DELSOL3

DELSOL3 is a moderately large program (approximately 9100 lines of code) accompanied by thorough documentation. The program reads a text input file and provides a text output file with nearly twenty pages of text information using the generic run settings. The output contains information on the power generation predictions, the system characteristics and parameters, heliostat dimensions, heliostat field layout, capital cost breakdown and much more.

DELSOL3 operates in two general modes. The first mode calculates the detailed performance of an existing system. The system could be one previously generated by DELSOL3 or one specified by the user based on other external models. The second mode is an optimization tool and takes a set of user inputs and iteratively identifies a system design capable of yielding the highest financial returns, accounting for capital and other costs against the projected electricity production. The focus of the optimization tool is the geometric relationship between the heliostat field and the central receiver. The tower height and receiver sizes are iteratively evaluated to determine the lowest cost based on a desired electricity output from the power cycle. The receiver surface is assumed to have convective and radiative losses that share a first-order relationship with

the overall surface area. Detailed receiver loss calculations on the scale of the current research are not done in DELSOL3.

The iterative process is worth examining, so the process flow-chart is presented in Figure 44 as it appears in the text of Kistler (1986). An important part of the iterative process is the initial guess range supplied by the user. As the range decreases in magnitude, the number of calculations required to achieve optimization decreases. This procedure requires some educated guess as to the probable tower height as a function of system design, and so difficulty may arise due to the implicit nature of the problem's solution.

The central strength of the DELSOL3 code is its versatility to handle many different system constraints and inputs. Examples include:

- Option to set economic parameters for plant optimization
- Option to choose a *surround* heliostat field, a north-based field, a user defined field (zone by zone), or an x-y coordinate defined field.
- Option to choose fixed, 2-axis tracking heliostats, or one-axis tracking heliostats on a rotation track about the central tower
- Option to specify geometric land constraints
- Option to select a plant oversize factor to accommodate thermal storage

In some cases, limitations can compromise the versatility of these options. For example, the user-defined heliostat fields must be symmetric in order to calculate daily or annual performance. This limitation may be an issue in a situation where a plant design that aims to produce more power during afternoon hours is desired (because of the increased value of the power during this time period), for example. Another example is that the user-defined field can only be used when calculating performance, and not when optimizing the system. However, despite some limitations, the code is remarkably versatile, option-inclusive, and well-documented and it provides an excellent foundation for work to be done.

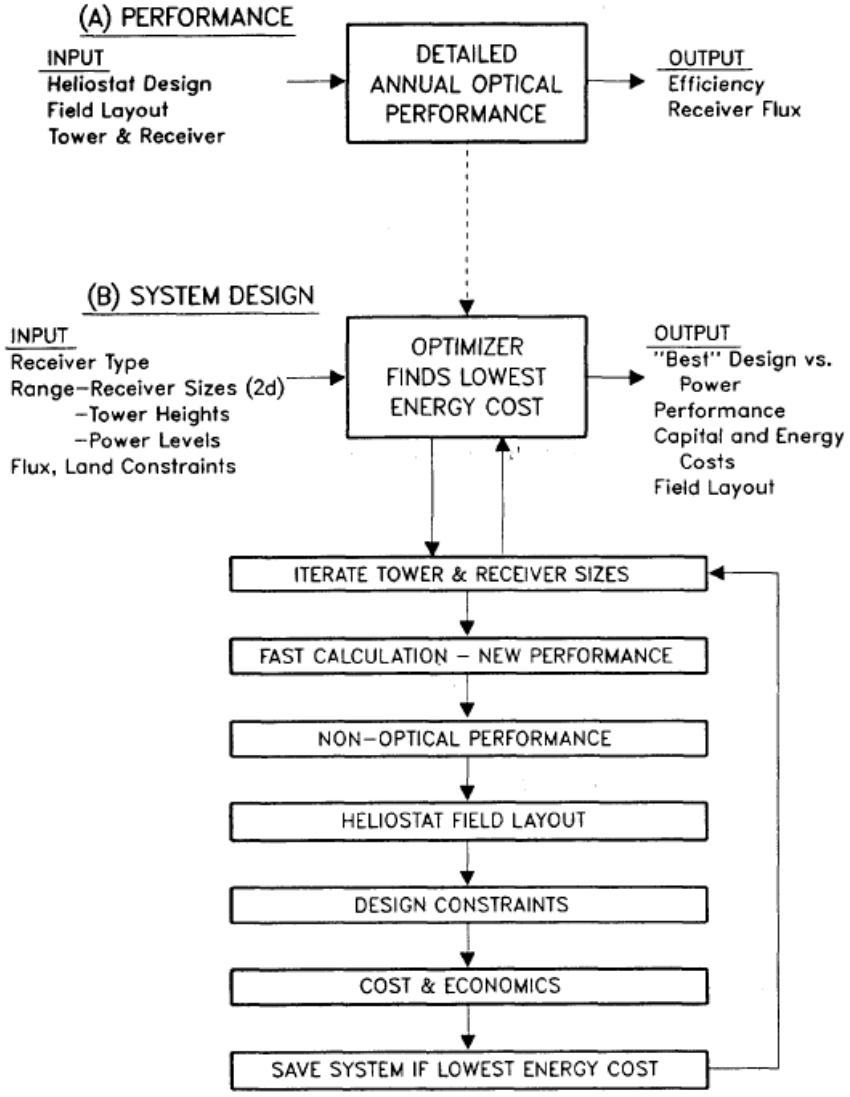


Figure 44: The two general operation modes for DELSOL3 code (Kistler, 1986).

Several significant challenges exist in integrating the DELSOL3 code into a larger structure. The original DELSOL3 code was constructed as a program – the top level in the Fortran hierarchy. Since program structures cannot easily be called by other programs, productive use of the code requires that it be available as a subroutine structure. Another problem in the original code was the use of the antiquated "Hollerith" data type which is not supported in current Fortran compilers. Additionally, The DELSOL3 code uses implicit variable declaration where groups of variables are assigned a data type based on the first letter of the variable name. This practice is confusing and

can easily lead to errors in memory allocation. DELSOL3 passes data between its subroutines and functions with the use of "Common" blocks, which assign values to variables based on variable order within the block. Errors can occur in memory management if integer variable types appear in the Common block before floating point variables. Finally, a large amount of output is printed to the console during program operation causing the code to run slowly.

Actions taken to remedy these problems and other minor issues are listed as follows:

- The code was restructured as a subroutine with no inputs or outputs; this process required the modification of exit procedures to match the requirements for a subroutine instead of a program.
- Hollerith strings were modified and updated to the current Fortran *string* structure.
- Common blocks were rearranged to store integer variables in sequence after more complex variable incidences.
- Unnecessary information printed to the output was eliminated muted.
- No implicit variable definition is allowed, as *implicit none* statements are added.
- Minor issues are also addressed:
    - An attempt to write information to a file that is never opened causes the program to hang. This code was removed.
    - In Fortran *Read* statements, item specifiers can be implied if the correct syntax is used. For example, statements such as *Read(5,REC)* can be used to read information from unit number 5 and namelist *REC*. To avoid ambiguity and possible error, the namelist specifier "*nml=*" was added such that the statement would appear *Read(5,nml=REC)*.

This undertaking resulted in an improved code that successfully compiles with modern Fortran compilers, including the Intel Fortran 10.1, Compaq Visual Fortran 6, and GCC Fortran (open source) compilers. Additionally, the code can be called from other programs and subroutines, and it runs more quickly since printed output is reduced or eliminated. The revised code also produces identical results to the original code.

5.2 The PTGen Implementation

The Power Tower Generator program (PTGen) consists of a graphical user interface and Fortran code to assist the user in generating the required input for long-term TRNSYS simulation of the power tower system. This package, developed as part of this research, can be bundled and distributed as a stand-alone program. PTGen uses the DELSOL3 Fortran code in its entirety, as well as several other Fortran programs and subroutines that were written for input file construction, output file formatting, and output data abridgment. The complete set of input data for long-term TRNSYS simulation requires multiple runs of the DELSOL3 code, with a unique input file for each run. Instead of forcing the user to modify the original text input files for DELSOL3 before each run, the PTGen program automates the process based on some simplified input criteria. The result is a much simpler, more time-efficient process for the user that still takes advantage of the strengths of DELSOL3.

The procedure that PTGen is designed to carry out begins with taking user input from the GUI and processing it into a formatted efficiency array for TRNSYS input. Figure 45 illustrates the steps involved.

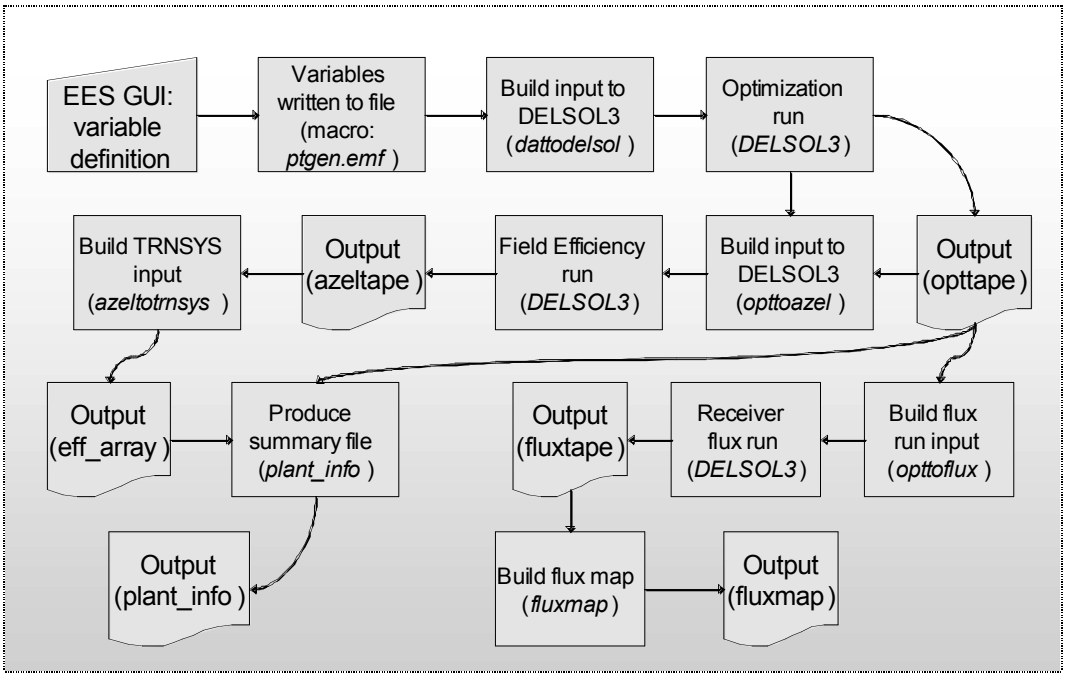


Figure 45: PTGen operation flowchart. Each process description is followed by the component name. Italic names denote program files, while plain type names represent text, data, or .csv file names.

A total of six original Fortran subroutines (dattodelsol, opttoazel, azeltotrnsys, plant_info, opttoflux, fluxmap) are called in this process, and DELSOL3 is called multiple times in three distinct capacities. The first call takes advantage of DELSOL3's ability to iteratively optimize a solar power tower plant based on user input set-points. The second and third calls are for performance calculations that use the plant sizing information produced during the optimization run. One particularly powerful feature of DELSOL3 is its capability to output an array containing the flux distribution provided by the heliostat field and incident on the tower receiver surface. The "receiver flux" performance run needs to be called many times, since the program is limited to outputting a single instantaneous flux map during each run.

This flux map is based on sophisticated aiming techniques that account for maximum flux levels based on the receiver material capabilities. The uneven flux distribution around the circumference of the receiver contributes to varying heat transfer losses at the surface, so the magnitude of these losses is an important consideration in the receiver

model. Figure 46 shows an example of the flux distribution incident on the receiver surface (in units of kW/m^2) at a given instant in time. The flux values are mapped as a function of normalized vertical height on the vertical axis (0 corresponds to the bottom of the receiver and 1 corresponds to the top) and circumferential position on the horizontal axis (180° represents the north facing portion of the receiver).

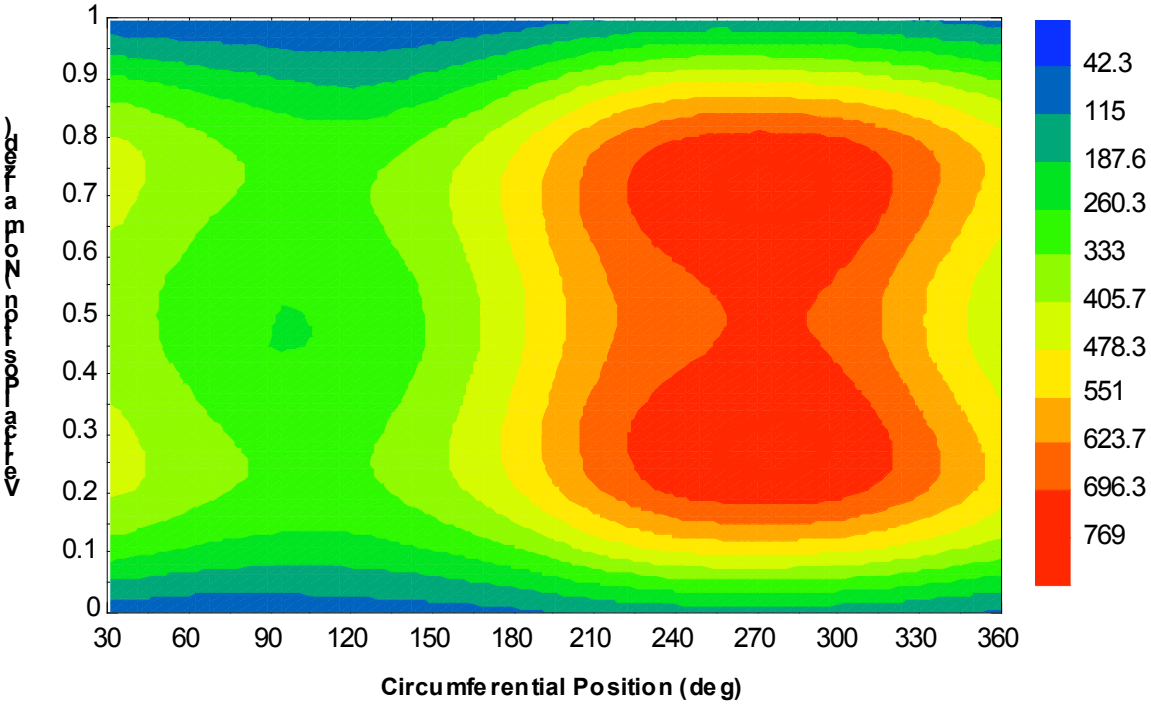


Figure 46: Flux on the receiver [kW/m^2] at a solar azimuth of 276° and a solar zenith of 68°. On the horizontal axis, 180° represents the north-facing portion of the cylinder

The implementation of these flux maps in the power tower simulation program requires careful consideration on several fronts. Primarily, it is important to know exactly what information these flux maps convey and how DELSOL3 goes about calculating them. Secondly, it is impractical and time-consuming to create a flux map for every single time step for use in TRNSYS during yearly simulations on hourly or shorter time steps. Rather, a finite number of flux maps capable of accurately representing the continuous distribution must be sought.

5.2.1 Flux Map Precision

An objective of this research was to use DELSOL3 to generate a series of flux maps that could be used to determine the energy flux on the tower receiver surface at any time during a year so that an annual simulation could be run. The intent was to develop capability to determine the energy flux on the tower as a function of solar position for a reference solar radiation value. This was accomplished by generating a number of flux maps for different solar positions and interpolating. The first significant challenge in determining the best way to implement the flux maps is choosing which flux maps are "representative." One might expect that the distribution would be hourly calculations for a certain number of days that are equally spaced throughout the year. Since the tilt of the earth's axis with respect to the sun provides a declination angle that is a sinusoidal function, evaluation would be useless beyond a period of π, or equivalently half of the year. Thus, the equally spaced distribution for only the six months between the winter solstice and the summer solstice were considered. Figure 47 shows the result of plotting the solar position (described by the zenith and azimuth angles) at equally spaced days throughout the half-year for the sunlight hours of the day.

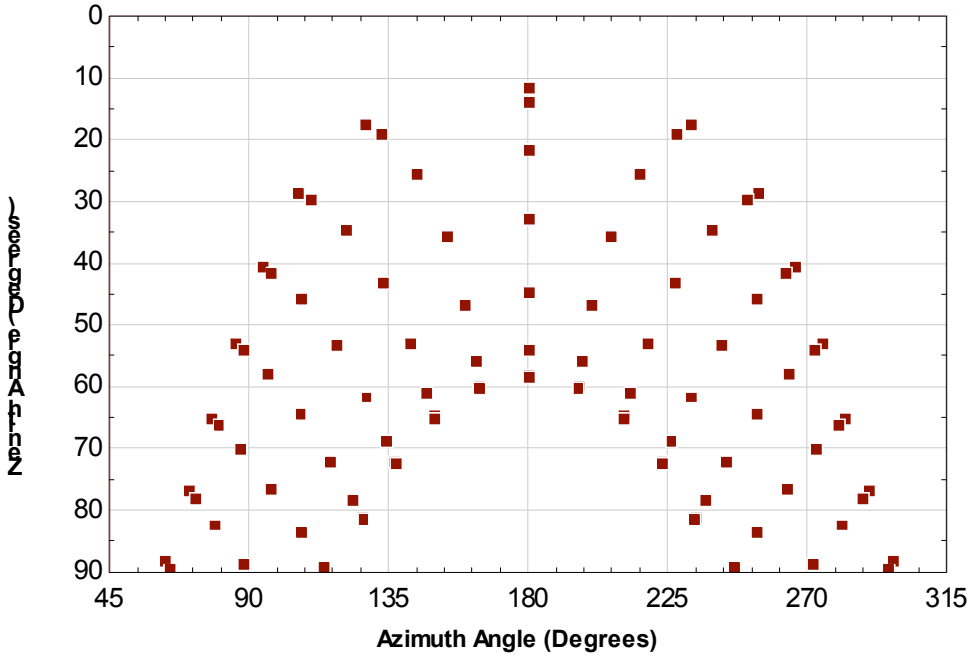


Figure 47: Solar position represented by equally spaced days throughout the half-year (180°=South).

This plot reveals the unequal spacing between the selected data points, which is a problem if we hope to achieve an even distribution that provides high accuracy for any solar position with relatively few flux map data points. Points tend to "bunch up" near the fringes of the plot area due to the decreased rate of change of the solar position during times near the solstices. To overcome this limitation, equal spacing was enforced between the solar declination angle instead of between the days of the year. By using the relationship shown in Eq.(5.1), the appropriate array of sample days was calculated.

$$Day_i = 355 - floor\left(\frac{\arccos\left(-1+2\left(i-1\right)\right)}{\left(nday-1\right)}\cdot\frac{\left(355-172\right)}{\pi}\right) \tag{5.1}$$

In this relationship, the number *355* represents the day of the year on which the winter solstice occurs, the number *172* represents the day of the year on which the summer solstice occurs, *nday* is the number of days that will be included in the selection which was chosen to be 8, as shown below, and the array is calculated for *i=1..nday*.

The resulting days of the year, according to this calculation where *nday=8*, are shown in Table 23. The adjusted plot of the selection of solar positions for flux map calculation is shown as Figure 48.

Table 23: Days of the year, spaced by declination angle

Value of *i*	Day of the year	Days after previous value
1	172	-
2	218	46
3	238	20
4	256	18
5	272	16
6	290	18
7	310	20
8	355	45

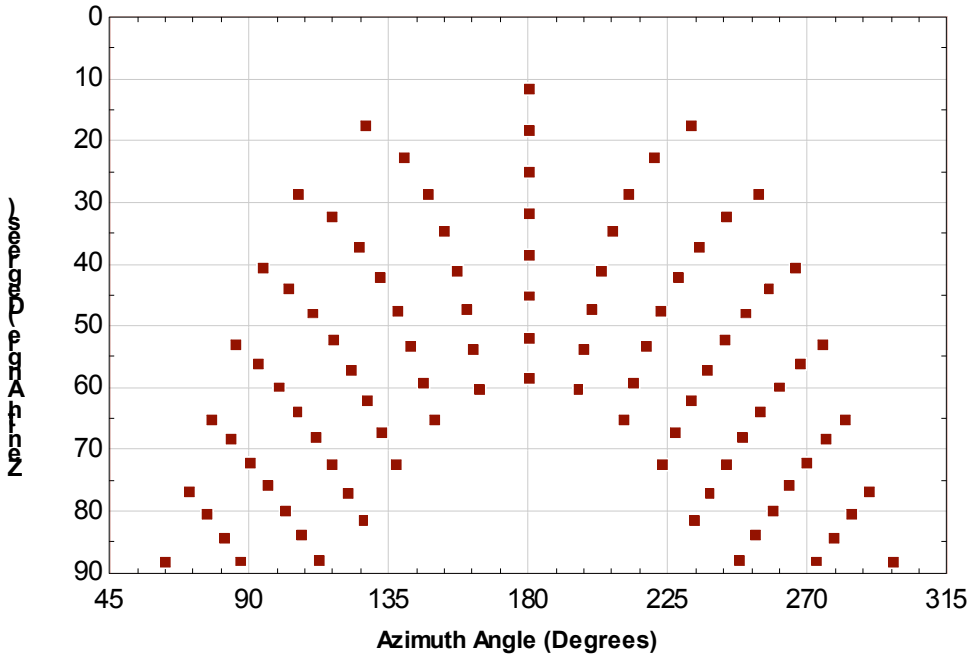


Figure 48: Solar positions used for the flux maps spaced equally by declination angle.

If flux maps are available for each point in Figure 5, TRNSYS can then interpolate them to estimate the actual flux distribution at any given instant in time. However, this raises the question of the required precision: how many flux maps will it take to accurately represent the yearly distribution? And, is it necessary to have a flux map for each point in Figure 5, or can a more coarse set of flux maps serve the same purpose?

To answer this question, a sensitivity study was undertaken that adjusted the number of flux maps, and then ran a yearly simulation with typical meteorological year (TMY2) weather data. The flux incident on the receiver at each hour was integrated over the year and the yearly total incident flux was compared in units of kW-hr/m^2 based on the number of flux maps in use. Figure 49 shows this yearly integrated flux as a 2-dimensional plot.

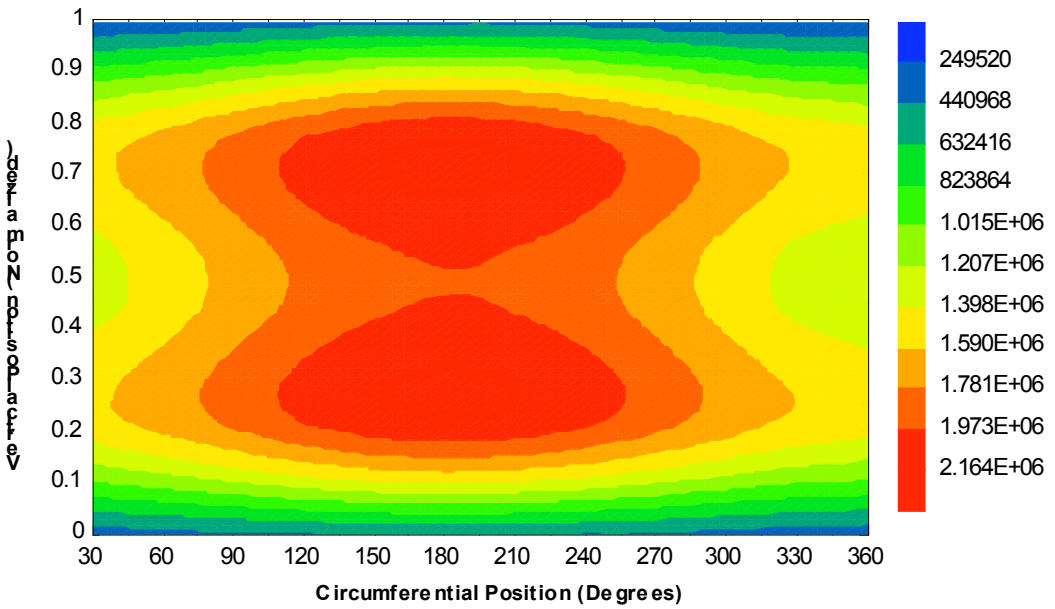


Figure 49: Yearly integrated flux on the receiver (kW-hr/m^2) as a function of azimuth and vertical position (m) for a 150 MW rated plant

The results from this sensitivity study show that using flux maps generated with 8 properly chosen days and interpolating the azimuth and zenith angles as necessary is adequate to estimate the yearly flux distribution. Compared to the calculated results the yearly flux with 16 days, and at every half hour, the errors were found to be less than 1% of the total flux. Table 24 shows a summary of the study results.

Table 24: Sensitivity Study results for determining the best number of flux maps

Number of Days	Hourly Frequency	Total No. of Flux Maps	Average Change [%]	Maximum Node Change [%]
4	Once/hour	48	0.123	0.525
8	Once/hour	96	-	-
10	Once/hour	120	0.024	0.491
12	Once/hour	144	0.033	0.365
8	Twice/hour	184	0.007	0.016
16	Once/hour	190	0.038	0.402
16	Twice/hour	368	0.047	0.391
4	Once/hour	48	0.123	0.525

This study demonstrates that the error associated with using 96 distinct flux maps to approximate the yearly distribution is appropriate and accurate to within at most

approximately 0.53%. By using fewer flux maps, and by allowing the program to simply choose the nearest flux map neighbor for calculations instead of interpolating at each solar position, the final TRNSYS simulations will run much more quickly.

5.2.2 Flux Map Normalization

To ensure consistency and predictability, all weather effects and atmospheric variations were uncoupled from the calculation of the flux maps. DELSOL3 outputs the flux maps based on a terrestrial insolation of 950 [W/m^2] regardless of solar position. Therefore, when using the flux maps in TRNSYS or any other simulation, the appropriate adjustments should be made, as shown in Eq.(5.2).

$$Adjusted\ flux_{i,j} = \left(Inst.\ flux\right) \times \frac{Map\ flux_{i,j}}{950\left[W/m^2\right]} \tag{5.2}$$

By dividing the area-specific receiver flux values ($Map\ flux_{i,j}$) by the adjustment factor of *(950 [W/m^2])* and then multiplying by the instantaneous terrestrial beam-normal insolation value reported as direct normal radiation (DNI) in a TMY2 data file (*Inst. flux*), the receiver flux map corresponding to actual radiation conditions can be calculated. This method is valid for receiver surface nodes $\left[i = 1..10,\ j = 1..12\right]$, which is the size of the DELSOL3 flux map (12×10 data points).

Although the method is presented for the two-dimensional flux array with indices *i* and *j*, previous analysis presented in the Central Receiver chapter indicates that the flux variation in the vertical direction can be condensed to a single average value without detrimental effects to the model. This simplification is reflected in the PTGen code, as the two-dimensional arrays are averaged and saved as a one-dimensional array (12×1).

5.2.3 Graphical User Interface (GUI) Overview

The GUI is built using the EES diagram window functions with imported graphics. Controls are separated into three general groups with the use of tabbed windows; the first

group contains general plant sizing requirements, plant location options, and a number of controls for file handling. The user can utilize these controls to allow manipulation of the DELSOL3 optimization input file or select options for viewing or deleting files generated by the program.

Because DELSOL3 includes a large number of variables, only the more commonly used variable controls have been included in the PTGen interface. A detailed analysis will often require the use of more DELSOL3 variables than have been included in the interface and this is made possible with the "Generate only the input file" and "Calculate w/ user edited file" checkboxes. For example, the modeler may want to include updated economic information in the plant optimization. To do this, PTGen can be used to set plant dimensions, guess values, or other information and generate the input text file for the plant optimization run. Instead of continuing on with the optimization, PTGen stops after the creating the input file, and the modeler can make additions or changes to the text file such that the pertinent economic information is included. PTGen can then optimize the plant based on the modified file.

In the case that the user wishes to manually alter the optimization input file before continuing with the program run, the following general procedure should be used:

- Select the plant size, heliostat and receiver dimensions as usual
- Select the "Generate only the input file" checkbox
- Click the "Calculate" button
- Navigate to the working directory and open the file titled "Input". Note that this file does not have a file-type extension, but it can be opened and edited with any text editor
- Make any changes to the file according to DELSOL3 guidelines. Save and close the file
- Deselect the "Generate only the input file" checkbox and select the "Calculate w/ user edited file" checkbox
- Click the "Calculate" button.

Figure 50 shows the first tabbed page in the PTGen GUI.

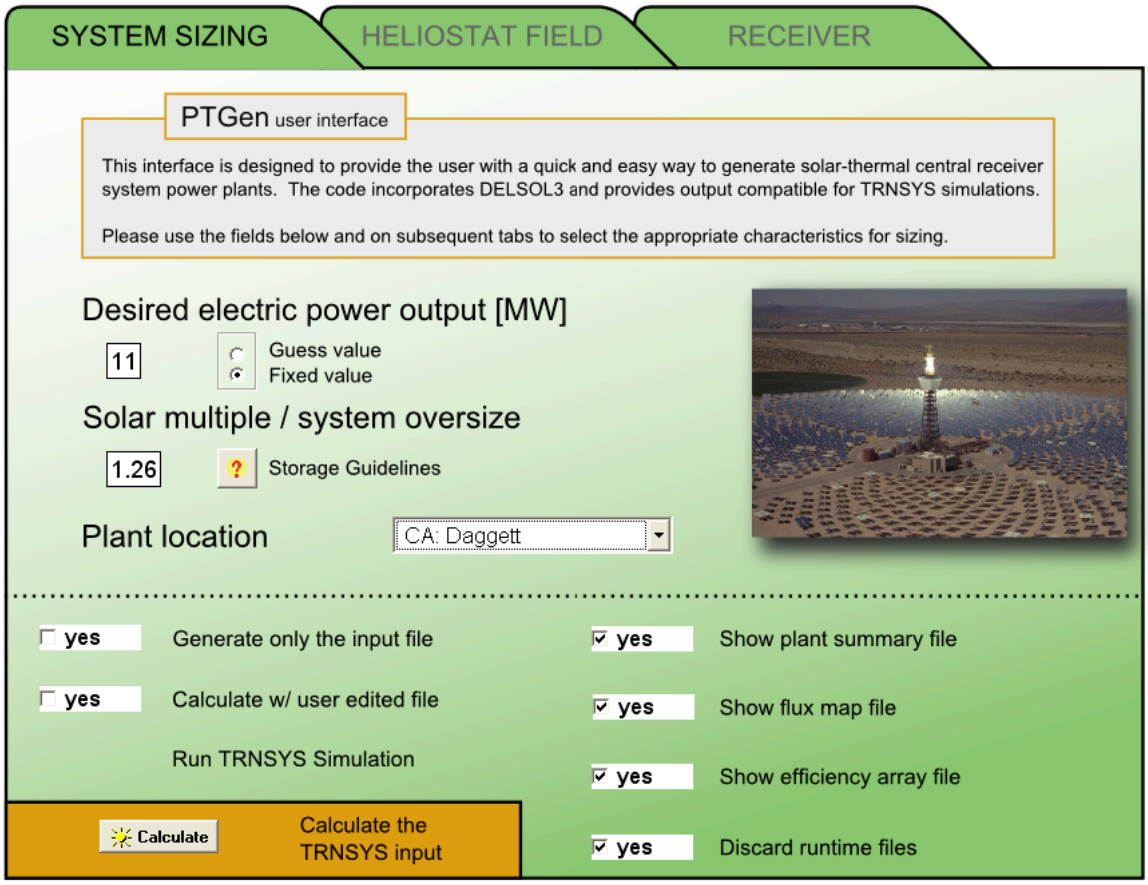


Figure 50: The first tab in the PTGen interface. This tab allows control of plant sizing and file-handling.

The system sizing tab also allows the specification of a system oversize factor, which is used to determine the amount of storage available. The factor is defined such that the plant is capable of producing electric power at the product of the nominal desired power and the oversize factor. In other words, for every one hour that the plant operates, it will produce thermal power to operate the electric power generation cycle for one hour times the oversize factor. In practice, the extra thermal power produced during the course of the day will be put into thermal storage where it can be dispatched as needed. For example, a 10*MWe* plant running at full power for 10 hours with an oversize factor of 1.5 will accumulate 5 hours of thermal storage to be used at the end of the day.

If optimization of the plant power with respect to other design parameters is desired, the "Guess Value" toggle can be selected. DELSOL3 will evaluate 20 nominal plant electric output powers at regular intervals between a minimum and maximum value as shown in Eq.(5.3).

$$\frac{1}{2}P_{nom,guess} \le P_{nom,guess} \le 5 \times P_{nom,guess} \tag{5.3}$$

The Heliostat Field tab is used to set properties relating to heliostat geometry and reflectivity. Overall heliostat height and width are specified here, and they can be elected as either default values or fixed values. If the default values are used, DELSOL3 will supply the programmed default values, which are a width of 9.91 *m* and a height of 9.93 *m*. If fixed values are used, DELSOL3 will accept the user-supplied values.

In addition to the overall size, the specific panel sizes can also be specified. Each individual panel in a heliostat can be assigned a size, and the spacing between panels can also be provided. Finally, specification of the heliostat field average mirror reflectivity value is available in this tab. This reflectivity value is included in the field efficiency data matrix as part of the overall field efficiency, so this loss should be assigned in PTGen since it is not an input to the heliostat field component in TRNSYS. Default values for the panels are shown in Table 25.

Table 25: Defaults for the heliostat cant panel settings (Kistler, 1986).

Item	Default
No. of vertical cant panels	8
No. of horizontal cant panels	2
Panel width [m]	4.88
Panel height [m]	1.22
Mirror reflectivity	0.91

Finally, the option of using round heliostats is available. In the case that round heliostats are selected, the diameter of the heliostat will be taken as the specified overall width, and the specified height will be ignored. Figure 51 shows the Heliostat Field tab.

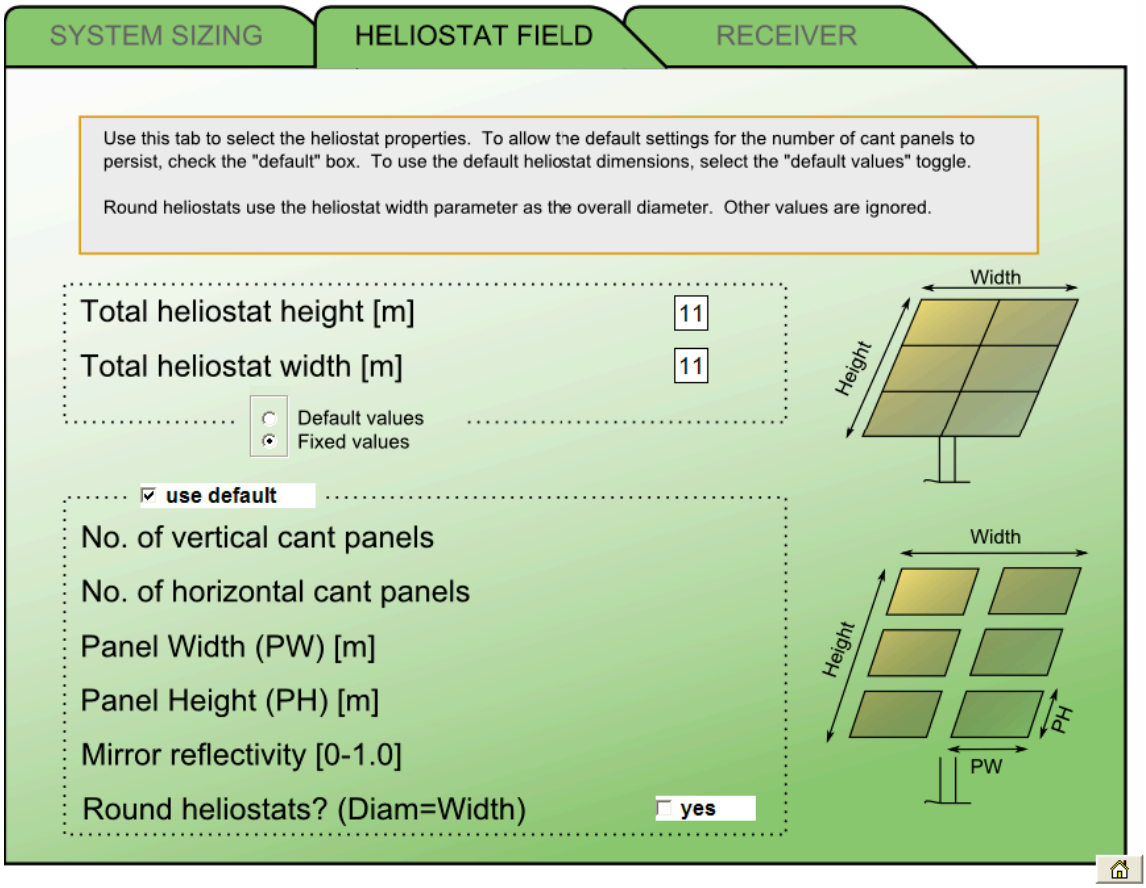


Figure 51: The Heliostat Field tab of the PTGen interface.

The final tab group contains options for the receiver, including both external and cavity receiver capabilities. A different set of options appears depending on which receiver configuration is chosen using the "Receiver Type" toggle. Several parameters are specified for both types of receiver, including the coating absorptivity, the maximum incident flux level, and the overall tower height. The default coating absorptivity for the receiver surface provided by DELSOL3 is 0.965 and the default flux limit is $600 kW/m^2$. Like the heliostat field mirror reflectivity, the tower absorptivity value is included in the heliostat field efficiency data matrix and should be specified using PTGen. To eliminate the possibility of accounting for the tower absorptivity more than once, the option of selecting tower absorptivity in TRNSYS is not included as an option.

The tower height is optimized according to the guess value provided, if optimization is desired. In this case, DELSOL3 performs calculations at 15 regular intervals between the range in Eq.(5.4).

$$0.6 \times H_{tower} \le H_{tower} \le 2 \times H_{tower} \tag{5.4}$$

For the external receiver, the receiver diameter and the panel height can be specified. These dimensions represent the geometry of the receiver surface, and not necessarily the receiver structure. If optimization of the receiver surface geometry is desired, the specified receiver height is ignored and 8 values of the receiver diameter in the range of Eq.(5.5) are calculated, each at 5 values of a height-to-diameter ratio in the range of Eq.(5.6). The "Receiver" tab with the external receiver option selected is shown in Figure 52.

$$0.4 \times D_{rec} \le D_{rec} \le 1.8 \times D_{rec} \tag{5.5}$$

$$0.6 \times \frac{H_{panel}}{D_{rec}} \le \frac{H_{panel}}{D_{rec}} \le 1.4 \times \frac{H_{panel}}{D_{rec}} \tag{5.6}$$

Additional geometry must be provided for the cavity-type receiver. Since the concave receiver surface is recessed inside a cavity, the flux from the heliostat field must pass through an aperture. The height and width of this rectangular aperture can be specified, as well as the radius of the receiver surface inside the cavity. For imaging purposes, the distance of the receiver cavity from the centerline of the receiver structure is significant, therefore, the structure width must be provided. Finally, the orientation of the cavity receiver must be provided in terms of an azimuth and zenith angle. For example, a cavity receiver whose aperture is oriented with a normal vector to the receiver surface pointing in the horizontal direction, the receiver zenith angle is 90°. If that same normal vector points to the North, the receiver azimuth angle is 180°. The cavity receiver geometry is shown in Figure 53, and the structure width is shown as ½ times the distance from the face of the cavity to the centerline of the structure.

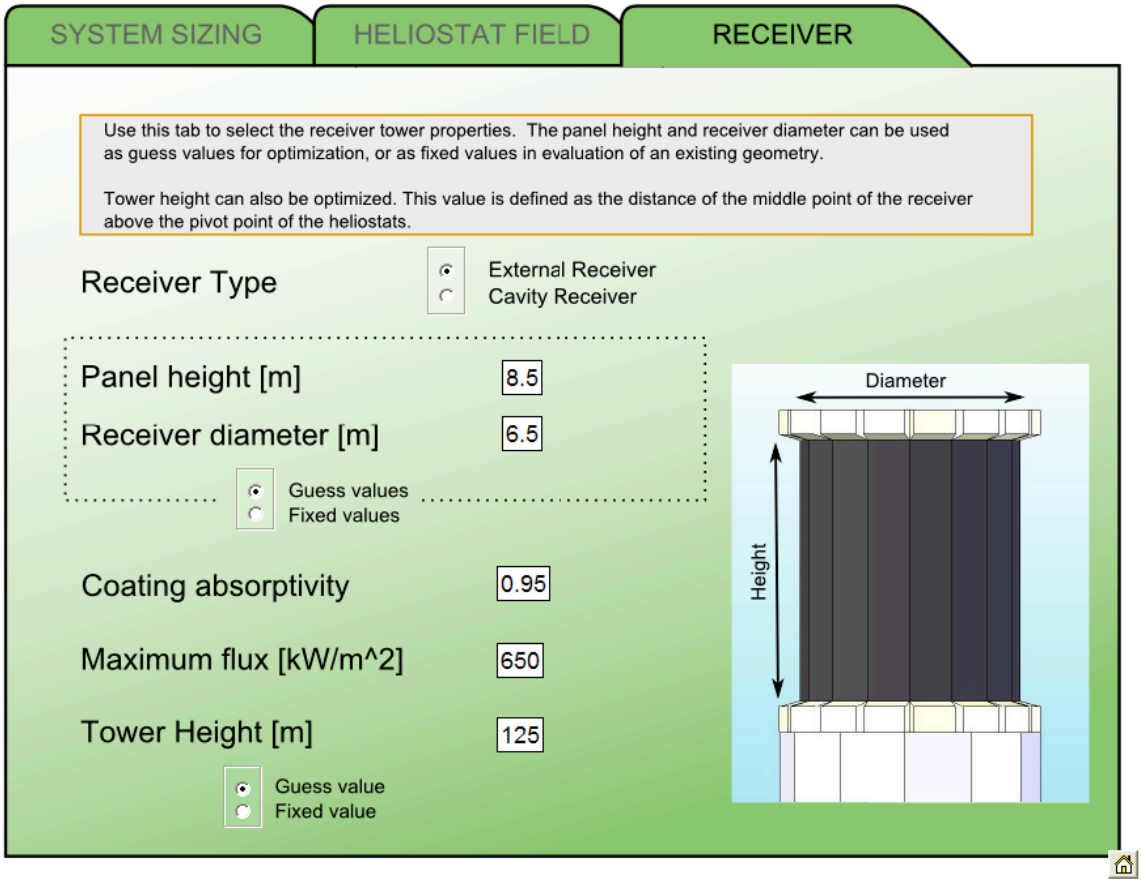


Figure 52: The Receiver tab of the PTGen program, with the external receiver option selected.

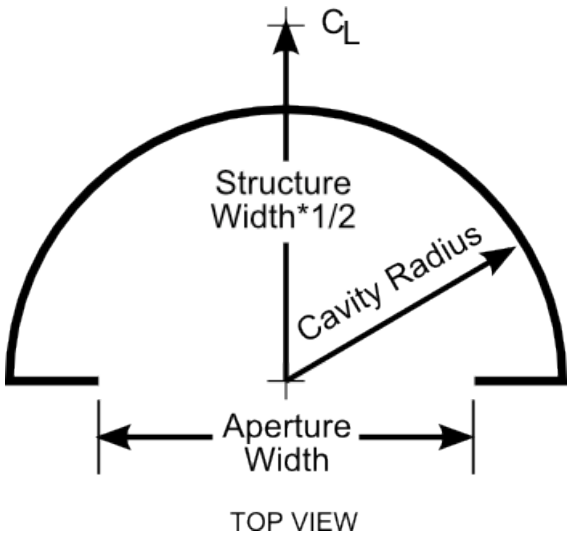


Figure 53: The cavity receiver geometry

In the case that the user wishes to optimize the cavity receiver geometry, a procedure similar to the external receiver optimization is used. Eight values of the aperture width are calculated in regular intervals in the same range as Eq.(5.5), where $W_{aperture}$ is substituted for D_{rec}. Five aperture height-to-width values are used in the same range as Eq.(5.6), where H_{panel} and D_{rec} are replaced by $H_{aperture}$ and $W_{aperture}$, respectively. The "Receiver" tab with the cavity receiver option selected is shown in Figure 54.

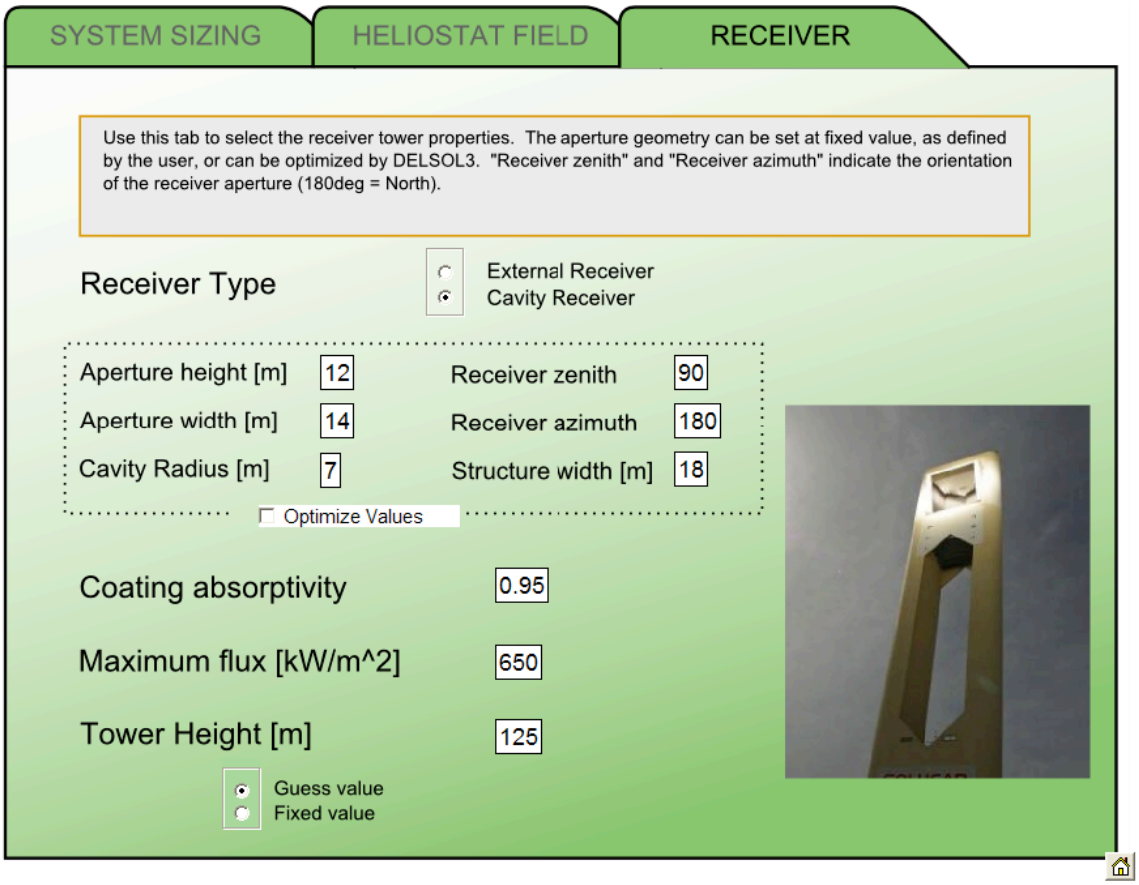


Figure 54: The Receiver tab of the PTGen program with the cavity receiver option selected

5.2.4 PTGen Program Output

A subsequent run of the PTGen program will produce three files. The first is a general plant summary file, called "plant_summary.txt". This file contains information read from DELSOL3 output, and does not reflect any additional calculations. Important

information contained in this file includes receiver geometry, heliostat field and heliostat information, and projected plant construction costs per kilowatt. It should be noted that the default DELSOL3 settings provide economic outputs in terms of 1984 dollars. These values can be adjusted by manually setting the economic parameters in the optimization input file, using the procedure outlined above in the *Graphical User Interface (GUI) Overview* section. Sample output for this file is shown for an 11 *MWe* plant in Daggett, CA. A plant oversize factor of 1.25 was used, providing roughly 1 hour of storage.

The first portion of the file provides information about the field layout and receiver geometry. For the sample system under evaluation, 702 heliostats at $9.82 \times 9.81 m^2$ were used. The best receiver tested was a $9.0 \times 9.0 m$ receiver at an elevation of 48.0 *m* above the heliostat pivot point.

```
*** Information file for a  11.MW plant ***
 ----------------------------------------

The number of heliostats is:   702
The size of each heliostat is:  98.41 [m^2]
The total mirror surface area is:    69081.0 [m^2]

 --- INFORMATION ABOUT THE RECEIVER:

The tower height is:  48.0 [m]
The receiver panel height is:  9.0 [m]
The receiver width is:  9.0 [m]
```

The annual energy breakdown of the plant includes information provided by DELSOL3 output, and provides values for the design power, overall conversion efficiency, total average hours of storage per day, total kilowatt-hours of electricity produced per year, and the capacity factor of the plant. It is important to reiterate that these values are not indicative of the long-term performance simulation results, but are only estimates provided by DELSOL3 based on the optimized plant performance over a short period of time. The overall efficiency is determined by accounting for field efficiency, receiver thermal losses, plant conversion efficiency, and parasitic losses. For more information on how these values are determined, refer to Kistler (1986).

```
--- ANNUAL ENERGY BREAKDOWN

DES.POWER  OVERALL EFFICIENCY  HRS.STORAGE  TOT.KWHR PER YEAR  CAPACITY FACTOR
(MWE)

11.00         0.199             1.31          0.2198E+08          0.228
```

The capital cost breakdown provides estimates for initial capital costs associated with construction of the plant. As previously discussed, the default economic parameters used to calculate these values are in terms of 1984 costs and can be modified. A percentage breakdown for the capital costs is included in this file. Table 26 provides a summary of each label. Finally, a cost in terms of dollars per kilowatt electric is provided.

Table 26: A summary of items provided by the DELSOL3 cost breakdown for projected power tower capital costs (Kistler, 1986).

Label	Description
LAND	Cost of the unimproved land area required for the plant, increased by 30% to account for roads and other additional surrounding land required, and increased by a fixed amount to account for improvements to the core land area of the plant.
HEL	Cost for the heliostats, including wiring and installation
WIRE	Additional costs for wiring, if not included in heliostat costs.
TOW	Costs associated with the tower structure. The cost scales with tower height. Towers taller than 120*m* are assumed to be concrete, while shorter towers are assumed to be steel.
REC	Cost of the receiver. The cost scales with square footage of surface area.
PIPE	Costs for piping from the power cycle/storage to the receiver, designed for molten salt in proportion to tower height and flow rate.
PUMP	Costs for pumps to move molten salt through storage, power cycle, and to receiver. Scale with flow rate and tower height.
STOR	Costs for thermal storage, including tank costs. If size reaches limit of 12,300 m^3, two equal volume tanks are used.
EPGS	Cost for the electric generating sub-system including the generator and turbine plant.
HTXCHG	Cost of the heat exchangers, scaled with thermal power of the plant.
FIXED	Assumed common field costs (buildings and roads, controls, etc.). Some costs scale with plant size.

```
--- CAPITAL COST BREAKDOWN

DES.POWER   DIRECT CAP.COST
(MWE)    (M$,CURRENT EST.)

11.00          37.55
PERCENT-
LAND    WIRE    HEL     TOW     REC     PIPE    PUMP    STOR    EPGS    HTXCHG  FIXED

  1.17    1.46   22.61    4.29   14.99    2.70    1.09    5.58   17.02    9.75   19.34
```

```
--- Economic figures of merit

 $/KWE (CURRENT $)
 4799.84
```

The second output file contains the heliostat field net efficiency as a function of solar position and it is called *eff_array.dat* (PTGen produces a file along with *eff_array.dat* called *array_view.csv* that also contains this information, but it is laid out in a more easily viewed 2-D matrix). This array contains 8×12 values, which indicate the overall efficiency values as solar azimuth and zenith angles vary. Total field efficiency is defined in Eq. (5.7), where the total radiation incident on the receiver is divided by the total radiation incident on the heliostat field mirrors for a given solar position.

$$\eta_{field,tot} = \frac{\dot{Q}''_{inc,tot} \cdot A_{rec,tot}}{I_{bn} \cdot A_{helio} \cdot N_{helio}} \tag{5.7}$$

In this equation, $\dot{Q}''_{inc,tot}$ is the total average flux incident on the receiver surface, $A_{rec,tot}$ is the area of the receiver surface, I_{bn} is the beam-normal solar radiation, A_{helio} is the area of one heliostat, and N_{helio} is the total number of heliostats in the field.

Figure 55 shows this field efficiency for an 11*MWe* external receiver. Since the heliostat field is more heavily distributed on the north side of the tower, efficiency values are higher when the solar position is in the southern sky relative to the tower. Optical losses also dominate when the sun is low in the sky, and this is reflected at zenith angles closer to 90°.

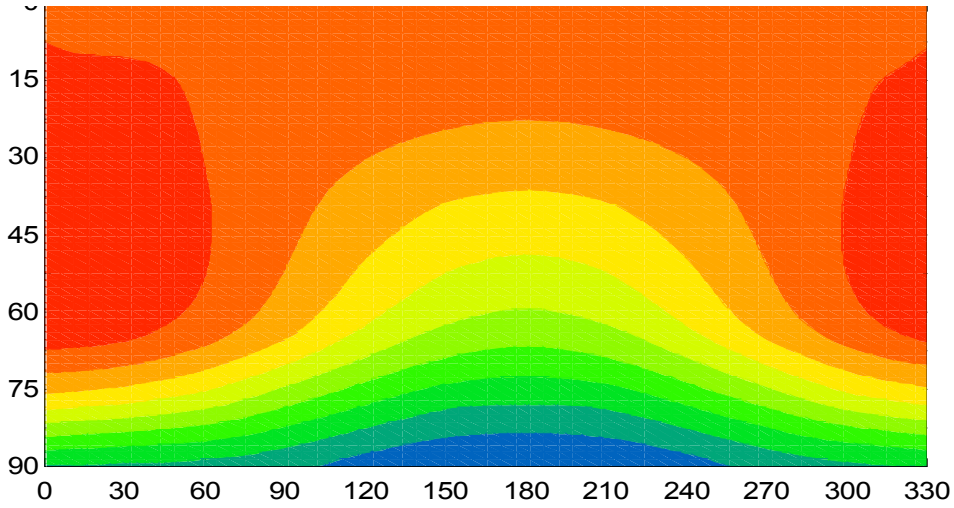


Figure 55: A field net efficiency map as a function of solar position.

The *eff_array.dat* file is used as an input to the heliostat field component in TRNSYS. The file is constructed to match the required format for this component, with the first two lines indicating the solar azimuth and zenith angles contained in the array, and the remaining lines indicating the net field efficiency at each point. The first two lines of the file are as follows, with all azimuth angle points listed first, and all zenith angle points listed second.

```
0.00,  30.00,  60.00,  90.00, 120.00, 150.00, 180.00, 210.00, 240.00,
270.00, 300.00, 330.00
0.50,   7.00,  15.00,  30.00,  45.00,  60.00,  75.00,  85.00
```

Efficiency values are then listed for each combination of points, with a value for each of the eight zenith angles in the progression of twelve azimuth angles. A total of 12×8 (96) values are provided. The first eight values for the azimuth angle of 0.00 and the eight zenith angles are shown here.

```
.63900
.64200
.64100
.64000
.63400
```

```
.60100
.47600
.30000
...
```

The third file produced by a PTGen run (*fluxmap.*csv) contains the flux distribution on the receiver for a number of solar positions. A discussion of the theory behind this output is provided in the *Flux Map Precision* section above. The flux map file is used as input for the central receiver component in TRNSYS, and it contains three types of information: the plant latitude, the solar position, day and hour of each flux map, and the vertically-averaged values of flux on the receiver. The first lines contain the header and plant latitude information.

```
This file contains the 2-D receiver flux data.

Plant lat.:, 34.9,deg
```

Next, an array of data provides the complete listing of the 96 solar positions included in the file along with the corresponding day and hour of the year used to generate the flux distribution. The data is listed in order according to day of the year, hour of the day (solar noon equals 0), solar azimuth angle (180°=South), and solar zenith angle (0°=Normal to the ground). A selected portion of these data points are shown.

```
172.0,  3.0,      266.1, 40.6,
172.0,  4.0,      274.6, 52.9,
172.0,  5.0,      282.1, 65.0,
172.0,  6.0,      289.6, 76.9,
172.0,  7.0,      297.5, 88.1,
218.0, -6.0,       76.3, 80.6,
218.0, -5.0,       84.3, 68.5,
218.0, -4.0,       92.8, 56.2,
218.0, -3.0,      102.8, 44.0,
218.0, -2.0,      116.7, 32.4,
218.0, -1.0,      140.0, 22.7,
218.0,  0.0,      180.0, 18.3,
218.0,  1.0,      220.0, 22.7,
```

Finally, the corresponding flux maps for each of these points are listed. The azimuth and zenith angle are provided, followed by the circumferential flux distribution. A selection of this data is shown.

```
Azimuth,Zenith
266.1, 40.6
Flux, by average [kW/m^2]
351.6,   338.5,   370.1,   407.9,   512.0,   636.9,   778.0,   872.4,
890.6,   815.6,   648.0,   456.4,


Azimuth,Zenith
274.6, 52.9
Flux, by average [kW/m^2]
382.5,   347.8,   334.0,   364.6,   431.0,   574.6,   761.0,   903.7,
936.1,   861.1,   691.0,   495.2,


Azimuth,Zenith
282.1, 65.0
Flux, by average [kW/m^2]
409.7,   366.6,   331.7,   315.3,   370.6,   516.2,   729.8,   933.2,
976.4,   903.9,   722.9,   515.3,
```

The information within these files generated by PTGen provides the required data for a full TRNSYS simulation. The plant geometry generated in the PTGen optimization run is used by the central receiver (Type 222), heliostat (Type 221), and storage (Type 4) components. The flux distribution file is used by the central receiver component, and the heliostat field efficiency matrix is used by the heliostat field component. Interaction between these components is managed by a plant controller component (Types 223 and 225). The following chapter discusses these components and their roles in plant simulation in detail.

6 The Plant Model in TRNSYS

An important result of this research is the integration of the heliostat field, central receiver, Rankine cycle, and storage models into the TRNSYS simulation environment. TRNSYS is a powerful tool that provides detailed analysis and design information for systems of interest by facilitating long-term transient simulations with relatively short time steps (on the order of one hour or less). Each of the previously developed autonomous component models can be arranged to interact within the TRNSYS environment, and the overall system behavior can be evaluated subject to a wide variety of operating conditions.

One of the CRS components that has not yet been discussed but remains an important consideration for future system designs is the thermal storage system. The variable nature of the solar resource and the mismatch between that solar resource and the utility grid demand for electricity create the need for some form of energy storage. Large scale electric energy storage is not a viable approach for storing the magnitude of electrical energy needed for achieving a meaningful shift of a central receiver plant electricity production for later use (DOE 2007). A potential viable alternative would be to store the equivalent energy in a thermal form.

A potential approach is the use of one or more tanks capable of storing heated molten salt for later use by the Rankine plant. Of concern with this storage medium is the risk of salt crystallization at low temperature points throughout the system (receiver, piping system, storage tanks, or power generation equipment). The integration of thermal energy storage provides an additional degree of freedom for the production of electricity but it also adds complexity to plant operating strategies. Considered in this section are alternative control strategies to dispatch the stored thermal energy appropriately.

Two of the most common storage techniques are addressed in this research. The first technique uses two approximately isothermal variable-volume tanks (one for hot storage

and one for cold storage). In this configuration, the fluid in the cold storage tank is pumped to the central receiver where heat is added, and the hot fluid then fills the hot storage tank instead of returning to the cold storage tank. The fluid in the hot storage tank is used to power Rankine cycle on demand, and the cold fluid returns to the cold storage tank. The resulting volume of heat transfer fluid in each tank varies as the dispatched power and available solar resource vary.

The second technique provides thermal storage using a single thermally stratified tank. The natural density variation of the stored fluid allows the warmer less dense heat transfer fluid to rise to the top of the tank and the cooler more dense fluid to fall to the bottom. A thin mixed layer called the "thermocline" separates the warm and cold layers in the tank. During a charge mode, the cooler fluid is taken from the bottom of the tank and pumped to the receiver and heated. The hot fluid is then returned at the top of the tank to maintain stratification. During discharge, the hot fluid is taken from the top of the tank and delivered to the power cycle. The cooler return fluid is then delivered to the bottom of the same tank to prevent destratification. The control strategies used to direct the plant for these two storage options are very different, and are discussed in detail in the following section.

When evaluating considerations such as the governing regulatory or incentive structure, the extent to which the plant is to be relied on as a base-load electric supplier, and the cost of the electrical generation equipment, another potentially desirable option is a backup fossil-fuel heat source. This additional complexity has been included in the control structure of both the variable-volume and stratified-tank control schemes.

This chapter presents the plant control schemes for variable-volume and stratified tank storage, and then takes a detailed look at the behavior of each plant configuration. Analysis is provided to demonstrate the model with specific case studies using the component models developed in this research.

6.1 The Central Receiver Plant with Variable-Volume Storage

A common practice in CSP plant design with thermal storage is to segregate the high temperature thermally charged heat transfer fluid from the cold fluid by the use of two separate insulated tanks. This was the case in the Solar II demonstration project, and continues to be of interest for CSP plant designs of both the power tower and parabolic trough technologies (Herrmann, et al. 2004). A plant configuration and control scheme were developed for this purpose, and the TRNSYS Simulation Studio arrangement is shown in Figure 56. The thermal storage was modeled using the existing Type 39 tank for variable-volume storage, and the auxiliary heat source was modeled using the existing Type 6 auxiliary heater component from the standard TRNSYS component library. The TRNSYS project file is presented in more detail in Appendix A and on the CD that accompanies this thesis.

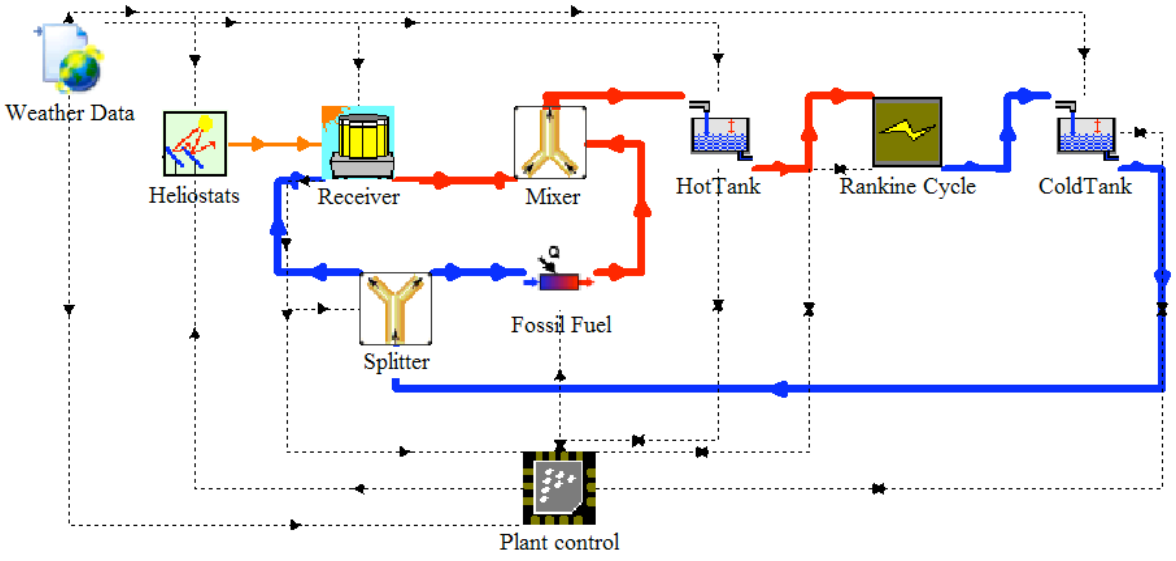


Figure 56: The TRNSYS plant configuration for a two-tank variable-volume system.

The central receiver system (CRS) plant control component was written to regulate the heat transfer fluid flows, power cycle operation, use of hybrid backup heat source, and heliostat field, ensuring that the operation falls within the boundaries of several user-specified parameters. A number of items can be used to constrain the operation of the power plant and manage the heat transfer fluid levels in the hot and cold storage tanks.

These items include:

- ***Hour of the day*** – to ensure the hour falls between the starting and ending hours specified for power cycle operation
- ***Solar zenith angle*** – to assist in determining the time at which the power plant should be shut down when storage is nearly exhausted
- ***Hot storage outlet temperature*** – to ensure that the temperature of the heat transfer fluid leaving the hot storage tank for the power cycle is maintained above a minimum level set by the user
- ***Cold storage outlet temperature*** – to ensure that the temperature of the heat transfer fluid leaving the cold storage tank for the heat source does not exceed a maximum temperature set by the user
- ***Load flow demand*** – to match the total flow from the heat source and from hot storage to the flow required by the power cycle
- ***The cycle standby period*** – in the case that the cycle enters standby mode due to insufficient flow from the heat source, the amount of time spent in this mode is monitored so that it does not exceed a limit set by the user
- ***Flow rate to the heat source*** – to ensure that the flow rate passing through the receiver does not exceed a maximum level set by the user
- ***Hot storage volume indicator*** and ***cold storage volume indicator*** – to check whether the fluid level in the storage tanks has fallen below or exceeded the minimum or maximum tank volume
- ***The hot storage fluid volume*** – to predict the remaining thermal resource required for standby mode

The modes of operation enacted by the controller are summarized in Table 27, with the mode of operation listed and followed by the operating criteria for that mode. Some modes of operation are initiated by more than one set of operating criteria.

Table 27: A summary of the modes of operation for the two-tank variable-volume control scheme.

Mode	Description	Criteria
1	The power cycle is turned off, and the heliostat field is defocused	The plant does not have a fossil-fuel backup, the hour of the day is not within the specified range, and the hot storage tank is completely full.
		The plant has fossil-fuel backup, the hour of the day is not within the specified range, and the hot tank is completely full.
2	The power cycle is turned off, and flow to/from the receiver is allowed	The plant does not have a fossil-fuel backup, the hour of the day is not within the specified range, and the hot tank has available volume for storage.
		The plant has a fossil-fuel backup, the hour of the day is not within the specified range, and the hot tank has available volume for storage.
3	The power cycle is on, but the field must be partially defocused	The plant does not have a fossil-fuel backup, the hour of the day is within the specified range, and the hot storage tank is completely full.
		The plant has a fossil-fuel backup, the hour of the day is within the specified range, and the hot tank is completely full.
4	The power cycle is on and flow from the receiver is supplemented with the backup heat source	The plant has a fossil-fuel backup, the hour of the day is within the specified range, the hot tank is not full, but the cold tank is completely full
5	The power cycle is on, flow to the cycle is fully met by flow from the receiver and/or flow from storage (Normal operation)	The plant does not have a fossil-fuel backup, the hour of the day is within the specified range, the cold tank is not full, and the flow from the receiver meets the power cycle flow demand
		The plant does not have a fossil-fuel backup, the hour of the day is within the specified range, the cold tank is not full, the flow from the receiver does not meet the power cycle flow demand, but the remaining volume in storage is projected to match the load
		The plant has a fossil-fuel backup, the hour of the day is within the specified range, the hot tank is not full, and the cold tank is not full
6	The power cycle is shut off for the remainder of the day, flow is allowed to the receiver	The plant does not have a fossil-fuel backup, the hour of the day is within the specified range, the hot tank is not full, but the cold tank is full (Note: the cycle is turned off for the day to avoid multiple full start-up/shut-down cycles in one day)
7	The power cycle is in standby	The plant does not have a fossil-fuel backup, the hour

	mode, operating at a fraction of the mass flow rate and without producing power	of the day is within the specified range, the hot tank is not full, the cold tank is not full, flow from the receiver does not meet the power cycle flow demand, and the projected required volume level is not met.

This arrangement is illustrated in Figure 57 with a flow diagram indicating the decision criteria for each mode of operation. The path of the diagram is selected by moving along the "*yes"* path if the criterion listed is true, and conversely along the "*no*" path if the criterion is false. Progress through the flow diagram continues until the end of the branch, where the mode of operation is indicated.

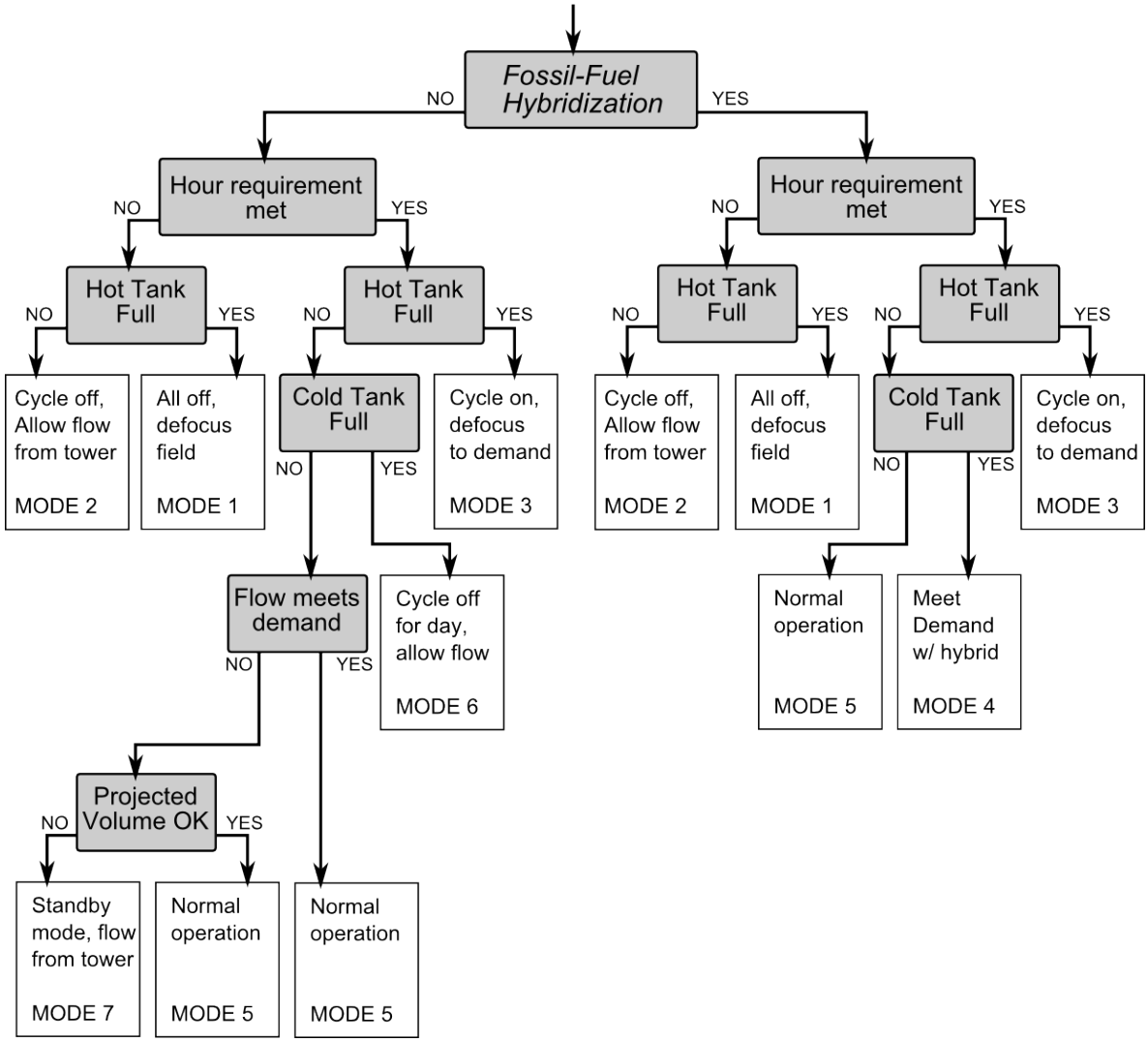


Figure 57: The plant control scheme for the two-tank variable-volume plant model.

This scheme is implemented using TRNSYS Type 223, an original component written for this research. The inputs and outputs are summarized in Table 28. This Type does not use any parameters.

Table 28: Summary of inputs and outputs for the TRNSYS CRS plant control component (Type 223)

Inputs			Outputs	
Name	*Default*	*Units*	*Name*	*Units*
Hour of the day	1	hr	Flow to power cycle	kg/hr
Start hour	12	hr	Flow to heat source	kg/hr
End hour	21	hr	Field Control	-
Solar zenith angle	10	degrees	Supplemental heat source control signal	-
Storage hot outlet temp	565	C	Pump on	-
Min temp to load	500	C	Cycle operation signal	-
Load flow demand	0	kg/hr	Mode	-
Storage cold outlet temp	290	C		
Max temp to heat source	350	C		
Flow from heat source	0	kg/hr		
Temp from heat source	0	C		
Hybridization mode	0	-		
Cycle standby period	2	hr		
Maximum flow rate to heat source	1000000	kg/hr		
Hot Tank Volume Level	0	-		
Cold Tank Volume Level	0	-		
Actual hot tank HTF volume	500	m^3		
Standby mode thermal fraction	0.15	-		
Reference HTF Mass flow rate	300000	kg/hr		
HTF fluid density	1850	kg/m^3		

6.2 The Central Receiver Plant with Stratified-Tank Storage

Although the two-tank model for thermal storage has been more commonly adopted in CSP plant designs, the prominence of thermal storage with stratification in solar applications has prompted the development of a single tank plant model. The potential benefits of using a single tank with stratification are reduction in required total tank volume, reduced construction costs due to consolidation from two tanks to one, a simplified control scheme, reduced piping, and reduced pumping requirements. However, there are some considerable drawbacks with stratification. Most notably, the

stratification is sensitive to inlet stream temperatures, the storage size must accommodate the large flow rates to avoid mixing, and axial conduction within the fluid can occur, reducing the temperature of the fluid passing to the power cycle. The plant design team must carefully consider these effects to determine their significance for each application, and the models presented in this discussion are designed to facilitate such an analysis.

The stratified tank plant controller operates much like the variable-volume plant controller; operation is restricted to be between a start and end hour of the day, the inlet temperature to the Rankine cycle and the inlet temperature to the receiver are controlled to be within user-specified limits, the option of a hybrid fossil fuel-assisted heat source is available, and a non-hybrid plant can operate in standby mode if necessary.

Figure 58 shows the TRNSYS Simulation Studio representation of the stratified-tank storage system with hybridization. The stratified tank was modeled using the existing TRNSYS library Type 4 component for stratified thermal storage. The flow of heat transfer fluid is indicated by the bold red (hot HTF) and blue (cold HTF) connectors. Feedback and control information is passed with dotted lines.

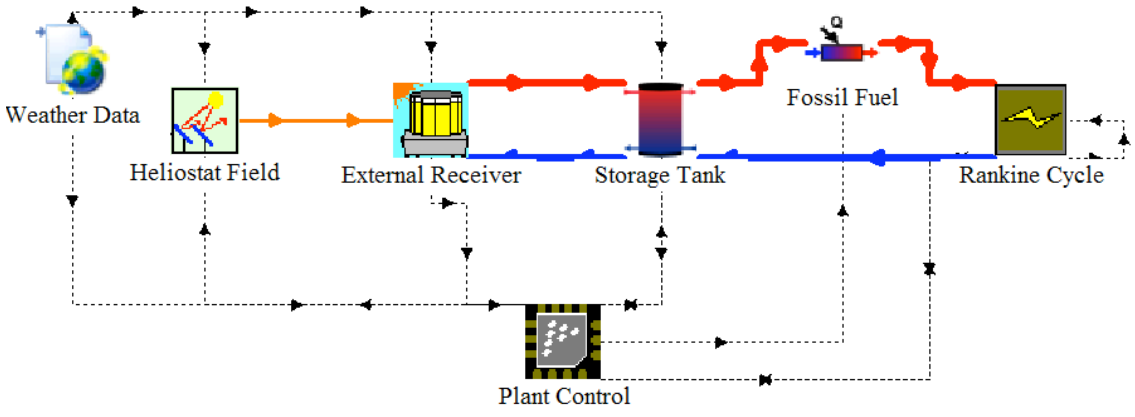


Figure 58: TRNSYS representation of the stratified tank plant model with hybridization.

The plant operates in six distinct modes that accommodate the possible hybrid heat source, tank charge levels, and user-specified temperatures and flow rates. Each mode is

summarized below in Table 29 with a brief description of the criteria required to reach each control decision.

Table 29: A summary of the modes of operation for the stratified tank control scheme.

Mode	Description	Criteria
1	The power cycle is on, flow to the cycle is fully met by flow from the receiver and/or flow from storage (Normal operation)	The hour of the day is within the specified range, the temperature of the heat transfer fluid at the power cycle inlet is above a minimum value, the temperature of the HTF at the tower inlet is below a maximum value
2	The power cycle is on, the tank is over-charged and the field must be partially defocused	The hour of the day is within the specified range, the temperature of the HTF at the power cycle inlet is above a minimum value, the temperature of the HTF at the tower inlet is above a maximum value
3	The power cycle is on, but the flow must be supplemented with hybrid heat addition	The hour of the day is within the specified range, the temperature of the HTF at the power cycle inlet is below a minimum value, the plant includes a fossil-fuel backup heat source
4	The power cycle is off, flow to the receiver is allowed	The hour of the day is not within the specified range, the HTF temperature at the tower inlet is above a maximum value
		The hour of the day requirement is within the specified range, the temperature of the HTF at the power cycle inlet is below a minimum value, the plant does not include a fossil-fuel backup heat source, standby mode is not available
5	The power cycle is off, and the field is completely defocused	The hour of the day is not within the specified range, and the tower inlet temperature is below a maximum value
6	The power cycle is in standby mode, flow to the receiver is allowed	The hour of the day is within the specified range, the temperature of the HTF at the power cycle inlet is below a minimum value, the plant includes a fossil-fuel backup heat source, standby mode is available

The flow diagram for this control scheme is illustrated in Figure 59. The plant controller for the stratified tank system has different inputs and outputs from the variable-volume system controller. Table 30 shows the parameters, inputs, and outputs for the TRNSYS plant controller (TYPE 225).

Table 30: Inputs and Outputs for the TRNSYS plant controller component (Type 225)

Inputs			Outputs	
Name	*Default*	*Units*	*Name*	*Units*
Hour of the day	1	hr	Flow to power cycle	kg/hr
Start hour	10	hr	Flow to heat source	kg/hr
End hour	23	hr	Field Control	-
Solar zenith angle	10	degrees	Supplemental heat source control signal	-
Storage hot outlet temp	565	C	Pump on	-
Min temp to load	540	C	Cycle operation signal	-
Load flow demand	0	kg/hr	Mode	-
Storage cold outlet temp	290	C		
Max temp to heat source	430	C		
Flow from heat source	0	kg/hr		
Temp from heat source	0	C		
Hybridization mode	0	-		
Cycle standby period	2	hr		

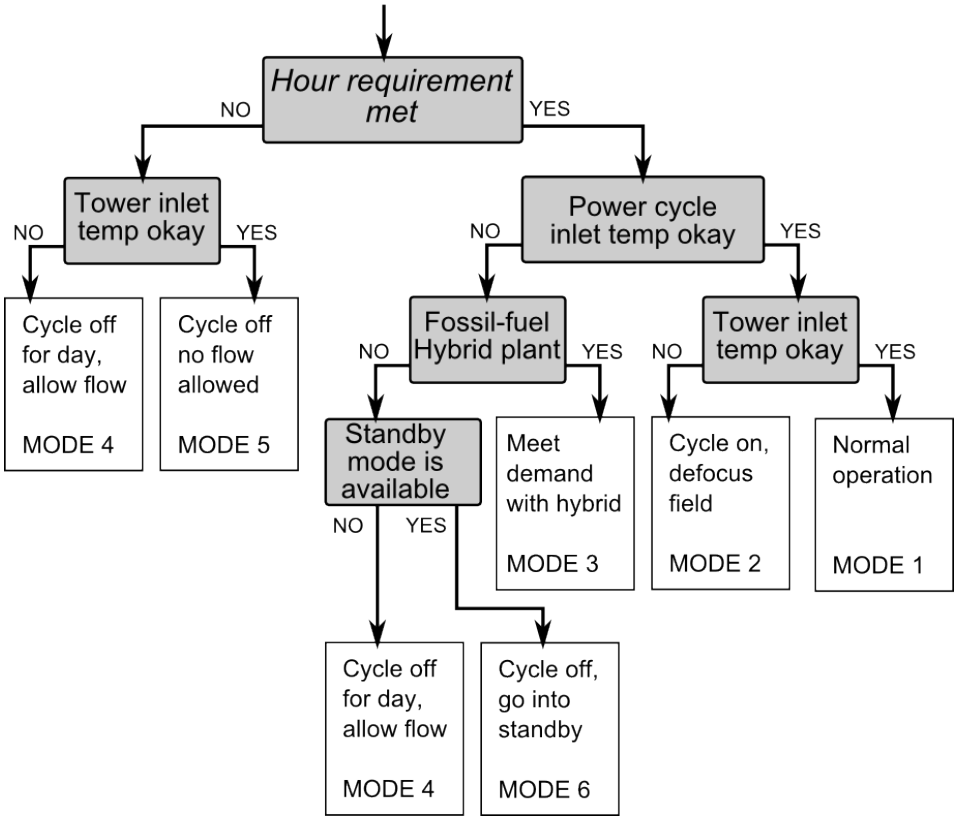


Figure 59: A flow diagram illustrating the control scheme for the stratified-tank plant.

Discussion in the following chapter provides a more detailed look at the behavior of these plant configurations under varying operating conditions. Additionally, these plant models will be used to perform analyses that exercise the component models and obtain conclusions about the nature of the CSP power tower technology.

6.3 Analysis Using the Plant Model

The purpose of the first analysis described in this section is to demonstrate the behavior of the two-tank variable volume storage model without hybridization (i.e., without use of supplementary conventional fuel). The analysis is based on an 11 *MWe* plant similar to the Solar II plant in Daggett, CA. The plant analysis begins by using the *PTGen* program to optimize a plant configuration that closely resembles the plant design of Solar II, and included in the parameter selection is an oversize factor of 1.30, representing a thermal energy generation capacity that is 1.30 times the nominal thermal energy generation requirement to meet the 11 *MWe* load. Table 31 summarizes the input values used in the PTGen interface.

Table 31: Summary of the inputs used for the PTGen portion of the analysis. Values not specified are assumed to be their default value.

Item	Value	Units
System Sizing Tab		
Desired electric power output (fixed value)	11.0	MW
Solar multiple / system oversize	1.30	-
Heliostat Field Tab		
Total heliostat height	6.0	m
Total heliostat width	6.6	m
No. of vertical cant panels	6	-
No. of horizontal cant panels	2	-
Panel Width	3.3	m
Panel Height	1.0	m
Mirror reflectivity	0.91	-
Receiver Tab		
Receiver type	External	
Panel height (guess value)	6.2	m
Receiver Diameter (guess value)	5.1	m
Receiver absorptivity	0.95	-
Maximum flux	800	kW/m^2
Tower height (fixed value)	76.2	m

Since the receiver absorptivity affects the overall heat absorption by the receiver, and the heat absorption is used to determine the best-optimized layout of the plant, the receiver coating absorptivity value must be provided to PTGen. Likewise, the heliostat field average reflectivity values must be provided to the PTGen interface for use in the DELSOL3 optimization run to ensure that the reflected energy from the field is accurately accounted for. The receiver absorptivity is set at 0.95 which is the absorptivity of the receiver coating used at Solar II (Pacheco, 2002), and the heliostat average reflectivity is set at 0.91 which is the default value used by PTGen (Kistler, 1986).

The resulting optimized plant was generated along with the efficiency and flux information. Because of the plant oversize factor, a modest 1.10 hours of estimated storage as determined by DELSOL3 has been included in the design to extend the operating hours of the plant and accommodate weather transients. The data summary file that was generated to provide feedback on the plant geometry and sizing is shown. As previously noted, the information that is provided for the annual energy breakdown shown below is an estimate based on cursory plant tests run by DELSOL3 during the optimization process. The detailed TRNSYS analysis to follow will provide more accurate characterization of the plant with calculations at short (sub-hourly) time intervals and integrated over a long period of time. Additional information on the plant summary file format is provided in Chapter 5.

```
*** Information file for a  11.MW plant ***
 ------------------------------------------

The number of heliostats is:  1583
The size of each heliostat is:  39.60 [m^2]
The total mirror surface area is:    62686.8 [m^2]

 --- INFORMATION ABOUT THE RECEIVER:

The tower height is:  76.2 [m]
The receiver panel height is:  6.2 [m]
The receiver width is:  5.1 [m]

 --- ANNUAL ENERGY BREAKDOWN

 DES.POWER  OVERALL EFFICIENCY  HRS.STORAGE  TOT.KWHR PER YEAR  CAPACITY FACTOR
 (MWE)

 11.00         0.216             1.10          0.2175E+08          0.226
```

```
 --- CAPITAL COST BREAKDOWN

 DES.POWER   DIRECT CAP.COST
 (MWE)    (M$,CURRENT EST.)

 11.00          38.26
 PERCENT-
LAND    WIRE    HEL     TOW     REC     PIPE    PUMP    STOR    EPGS    HTXCHG  FIXED

  0.77    1.46   20.23    5.25   16.51    4.28    1.17    4.85   16.70    9.57   19.21

 --- Economic figures of merit

 $/KWE (CURRENT $)
 4890.15
```

This optimization was performed with the economic defaults provided by DELSOL3 which are supplied in 1984 dollars. The economic and cost estimates used by DELSOL3 are based on manufacturing techniques and material costs around the time of the program's release. Recognizing that the output from DELSOL3 are more than two and a half decades old, simply adjusting the costs for inflation will improve but not completely clarify the economic validity of the results since it is likely that other improvements to the manufacturing processes and materials used in plant construction have occurred during this timeframe. This realization will lead to somewhat different system optimization based on economics. Thus, this analysis is intended as only illustrative, and a thorough economic review is required to accurately assess the viability of a current "state-of-the-art" power tower system. For example, the Spanish PS10 project provides 11 *MWe* (gross) power at a total investment cost of approximately 35M€ in 2006 currency, which translates to approximately $3,600/kWe using the 2006 exchange rate of $0.83/1.00€ (NREL, 2007).

In addition to the plant summary file, optimization with *PTGen* produces a data matrix containing the total field efficiency as a function of solar position. Figure 60 shows the array plotted as a function of solar azimuth and elevation angles.

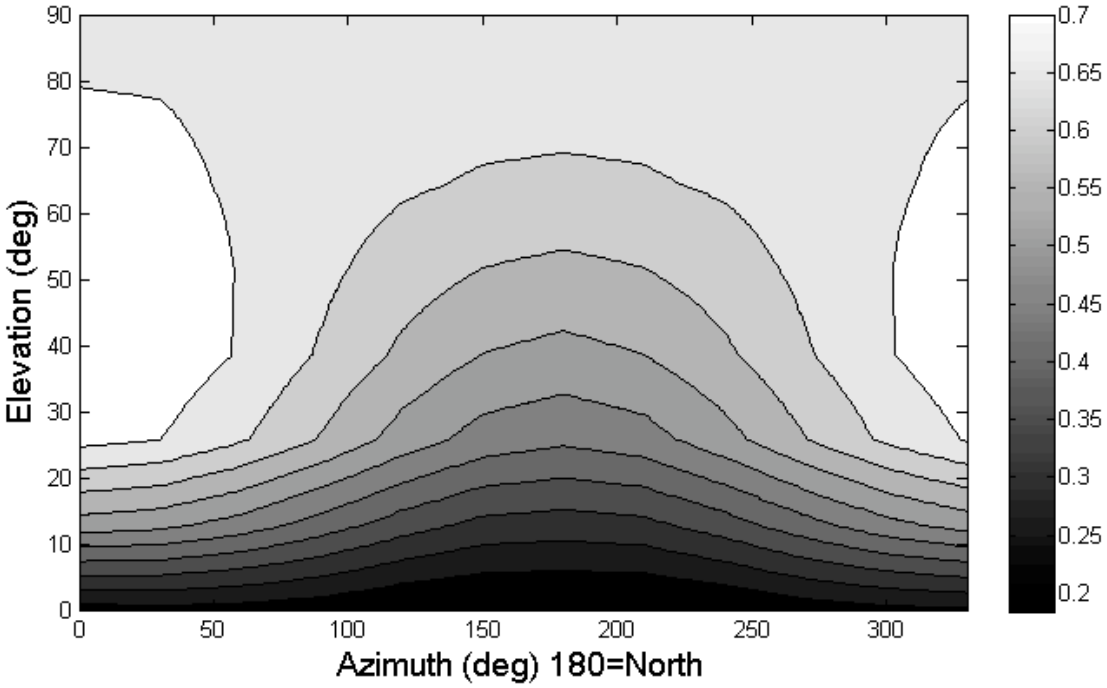


Figure 60: The net field efficiency as a function of solar position for the current analysis.

The efficiency file mapped in Figure 60 contains the information needed for the heliostat component in the TRNSYS model (Type 221). The heliostat component uses the array to determine the net field efficiency as a function of solar position for the specific plant design under evaluation. The flux map file – also produced by *PTGen* – is used by the receiver component (Type 222). The information produced from this preliminary plant sizing operation will be used for both the two-tank model and the single stratified tank model simulations discussed below.

6.3.1 Analysis with the Two-Tank Model

The plant arrangement for the two-tank variable volume model is described in section 6.1. Table 32 summarizes the important input values and options selected for this analysis (A more complete listing of the simulation is provided in Appendix A).

Table 32: Summary of the inputs for the plant TRNSYS two variable-volume tanks simulation.

Component	Role	Item	Value	Units
Type 15	Weather data	External File	US-CA-Dagget-23161.TM2	-
Type 221	Heliostat field	External Files	eff_array.dat	
Type 222	Receiver	Number of panels	24	-
Type 222	Receiver	Receiver diameter	5.1	m
Type 222	Receiver	Panel Height	6.2	m
Type 222	Receiver	Tower height	76.2	m
Type 222	Receiver	Coolant	12	-
Type 222	Receiver	External Files	fluxmap.csv	
Type 39	Hot Tank	Tank operation mode	1	-
Type 39	Hot Tank	Overall tank volume	1000	m^3
Type 39	Hot Tank	Minimum fluid volume	100	m^3
Type 39	Hot Tank	Maximum fluid volume	900	m^3
Type 39	Hot Tank	Initial fluid temperature	565	C
Type 39	Hot Tank	Initial fluid volume	500	m^3
Type 39	Cold Tank	Tank operation mode	1	-
Type 39	Cold Tank	Overall tank volume	1000	m^3
Type 39	Cold Tank	Minimum fluid volume	100	m^3
Type 39	Cold Tank	Maximum fluid volume	900	m^3
Type 39	Cold Tank	Initial fluid temperature	290	C
Type 39	Cold Tank	Initial fluid volume	500	m^3
Type 224	Rankine Cycle	Reference Power	11000	kW
Type 224	Rankine Cycle	Reference efficiency	.3400	-
Type 224	Rankine Cycle	Demand Variable**	11000	kW
Type 223	Plant Control	Start Hour	10	Hr
Type 223	Plant Control	Hybridization mode	0	-
Type 223	Plant Control	End Hour	23	Hr
TRNSYS input file is "Plant Model_two tanks.dck" **Demand variable indicates desired power produced by Rankine cycle				

The simulation is run for one year at half-hour time-steps using the weather data for a typical meteorological year (TMY2) in Daggett, CA, and several outputs are monitored. By computing the sum of the radiation power incident on the field and the total power produced over the course of the year, several plant characteristics can be determined by integrating the short-term results. The total thermal power incident on the 61,617 m^2 heliostat field over the course of the year is calculated in Eq.(6.1).

$$\int_{year} I_{bn} \cdot A_{helio} \cdot N_{helio} dt = 172.00 \left[GWth \cdot hr \right] \tag{6.1}$$

For the plant under consideration with 172.00 *GWth-hr* of available radiation per year, the following are determined:

- The total yearly generated electric energy is **36.73 *GWe-hr*** and the total thermal energy produced by the receiver is **99.46 *GWth-hr***
- At 11 *MWe* rated capacity, and if the plant were to operate at the 11*MWe* nominal power level for the duration of the 8,760 hours in a year, the maximum total yearly energy delivery capacity is **96.37 *GW-hr***
- The plant capacity factor (or the ratio of the amount of electricity produced compared to the integrated capacity for production presented in the previous list item) is **0.381**
- The total conversion efficiency from solar radiation to electric power is on average for the year is **21.36%**. Note that this does not include parasitic losses from pumping, the heliostat field, or facility operation.
- The average receiver thermal efficiency (not including reflective losses from the surface or pumping losses) is **93.14 %**
- The total annual energy required to power the receiver pump is **0.161 GWe-hr**

A closer look at the plant simulation reveals behavior that might be expected. At times when the heat transfer fluid flow rate to the receiver exceeds the flow rate to the power cycle, the hot storage tank volume increases and the cold storage tank volume decreases. The volume of the fluid in hot storage tank plus the volume of the fluid in the cold storage tank is always equal to a constant (in this case, 1000 m^3). Figure 61 shows the volumes of the hot and cold fluids in the storage tanks, the power cycle heat transfer fluid (HTF) flow rate, and the receiver HTF flow rate for a selected period of time in mid-June.

Also of interest is the thermal performance of the receiver during this time period. The receiver thermal efficiency and the overall receiver efficiency are shown in Figure 62. The overall receiver efficiency accounts for losses due to reflection and pumping.

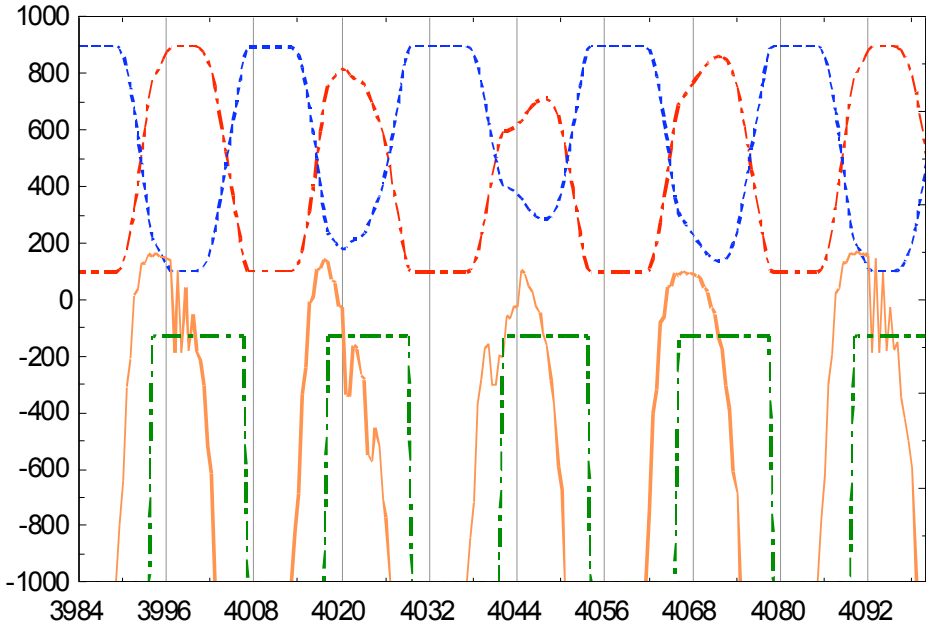


Figure 61: The hot and cold tank volume levels (left axis) and the heat transfer fluid flow rates to the power cycle and the receiver (right axis) for June 14-18th.

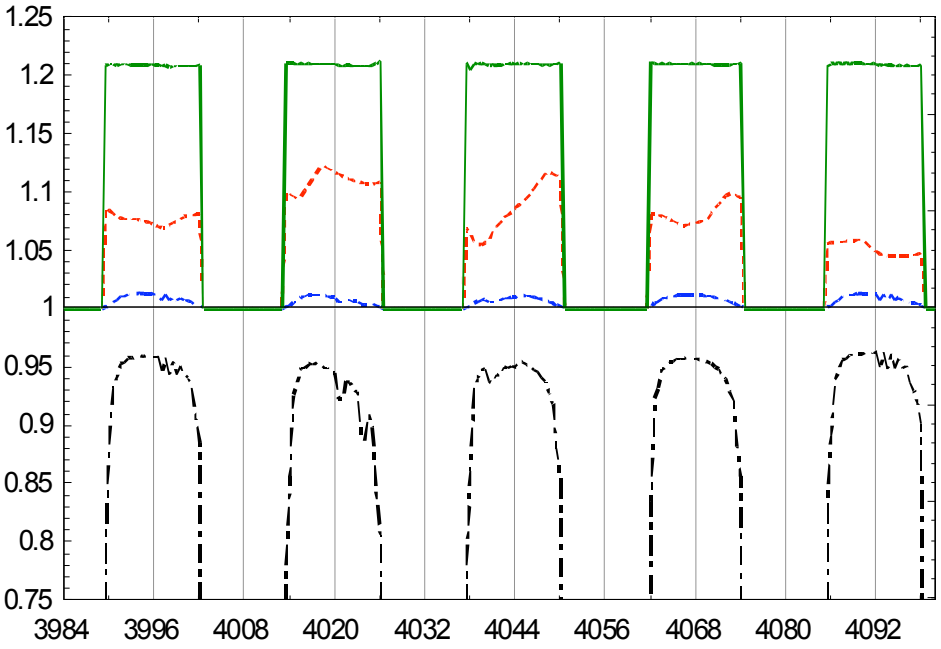


Figure 62: Receiver efficiency (left axis) and receiver thermal losses and pumping power (right axis) for June 14-18th.

The minimum heat transfer fluid level allowed in the hot tank is 0.9 *m,* or approximately 100 m^3; once the hot tank volume reaches the minimum specified volume, the power cycle shuts down. This effect is observed in Figure 61 where the flow rate to the Rankine cycle goes to zero as the hot tank volume reaches the minimum value. This model also demonstrates the capability of a limited amount of thermal storage to shift the electricity production period from the time of solar availability to the time of peak demand. Although the cycle power production doesn't begin until 10:00am, the cycle is able to run into the evening after the solar resource is no longer available, and thus can match the grid peaking demand more closely. Were a larger thermal capacity added to the plant and the storage size increased, the plant could become even more like a base-load power supply. However, the economics of such a plant design may or may not be favorable.

This plant configuration can only provide power production only when the solar resource is available or if there is sufficient energy stored in the hot fluid storage tank. If neither of these conditions is met, the plant must be shut down. This can be problematic if power production is needed. For example, during the winter months, the solar resource is not as strong as during the summer months. Figure 63 shows an example of typical operation during the winter months for the two-tank plant configuration discussed above.

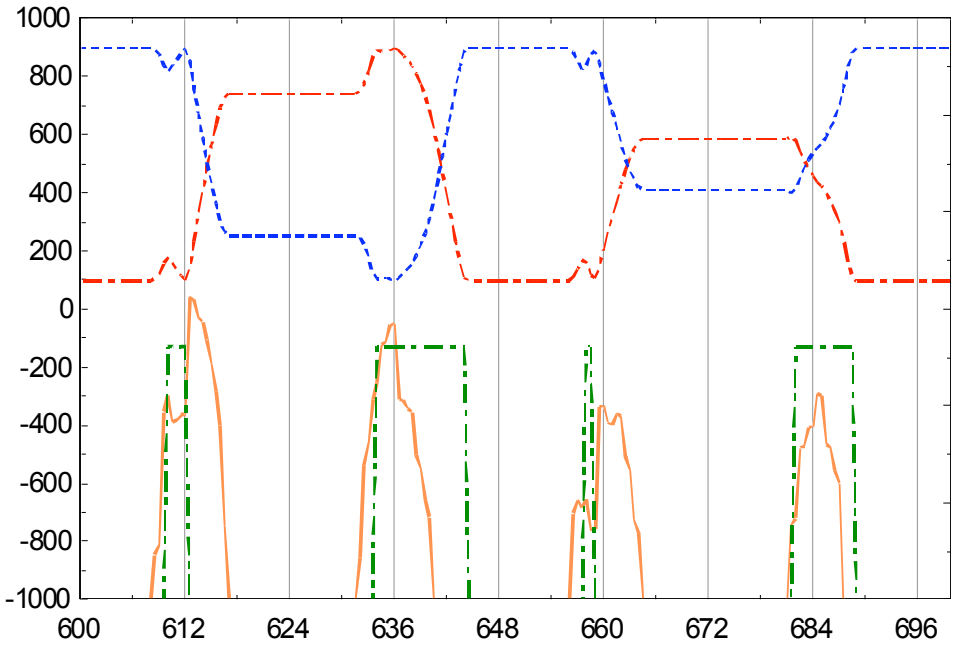


Figure 63: The hot and cold tank volume (left axis) and the HTF flow rate to the Rankine cycle and receiver (right axis) from January 25-28th.

The electricity production is truncated to accommodate the reduced solar resource availability during the selected time period near the end of January. In cases where a more firm power generation source is required, the use of a fossil-fuel backup system is a possibility. To evaluate the effect of adding a hybrid capability to the plant, the previous analysis is repeated using identical conditions, except that the hybrid option has been enabled. The effect of this option is that the power cycle will run regardless of solar availability between the specified hour range (10:00 am to 11:00 pm, in this case) and the supplemental heat source will be used to provide thermal energy when the solar resource is not sufficient. The results of this analysis are summarized as follows:

- The total yearly generated electric power is **52.20** ***GWe-hr***
- The total thermal power produced by the receiver is **99.83** ***GWth-hr*** and the total thermal power produced by the fossil-fuel backup is **41.35** ***GWth-hr***
- The fraction of the power produced by the solar source (the solar fraction) is **70.71%**

- The plant capacity factor (or the ratio of the amount of electricity produced compared to the capacity for production) is **0.542**

The plant capacity factor when a backup system is in place depends on the desired time period of operation and the amount of thermal storage capacity in the system. If the simulation had been run with an ending hour of 10:00 pm instead of 11:00 pm, the solar fraction would instead have increased since the required operation time period decreased. As expected, the fuel backup system creates a steady and predictable electricity generation pattern, and the relationships shown in Figure 63 are again plotted with the hybrid backup system enabled in Figure 64.

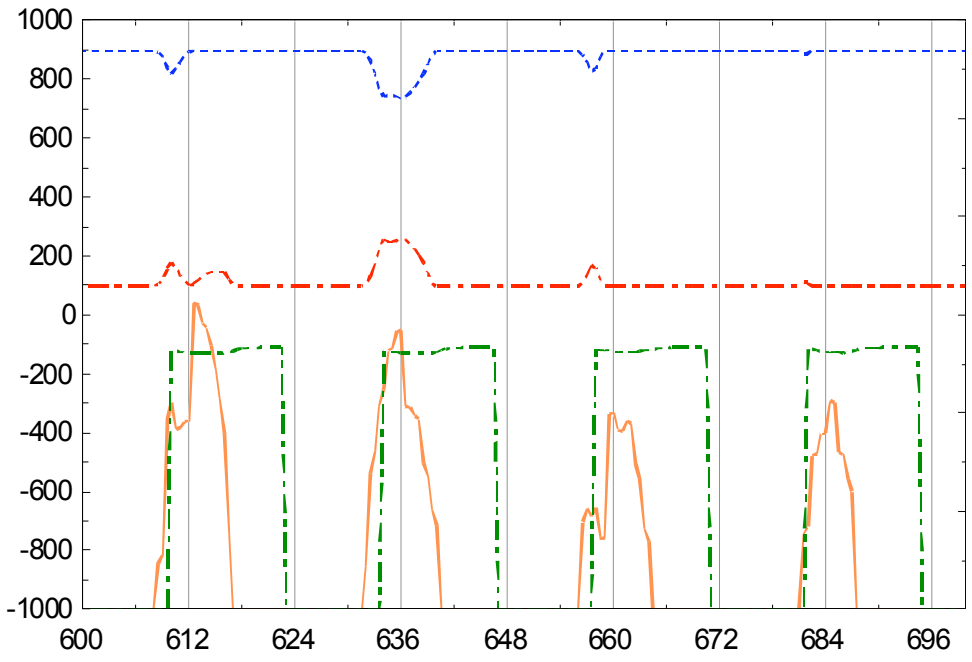


Figure 64: The hot and cold tank volume (left axis) and the flow rate to the tower and the Rankine cycle (right-axis) for the hybrid backup system from January 25-28th.

The plot in Figure 64 demonstrates that even in times of limited solar resource, the plant with a fuel backup can still be used to produce electricity to meet the demand. Although this arrangement may be ideal for the customer (namely the utility), state or federal

subsidy requirements may limit the implementation of fossil-fuel based backup systems to require a certain solar fraction for electricity production.

6.3.2 Analysis with the Stratified Tank Model

Using the optimized plant layout from the previous analysis, the single stratified tank model can be substituted for the two-tank variable volume model. A single stratified tank can presumably be installed at lower cost than the two-tank system considered in the previous section. The inputs for this plant model are summarized in Table 33, with the default and non-essential variables omitted. A more thorough listing of this information along with the TRNSYS project file is provided in Appendix A.

Table 33: Summary of the inputs for the TRNSYS stratified tank plant model.

Component	Role	Item	Value	Units
Type 15	Weather data	External File	US-CA-Dagget-23161.TM2	-
Type 221	Heliostat field	External Files	eff_array.dat	
Type 222	Receiver	Number of panels	24	-
Type 222	Receiver	Receiver diameter	5.1	m
Type 222	Receiver	Panel Height	6.2	m
Type 222	Receiver	Tower height	76.2	m
Type 222	Receiver	Coolant	12	-
Type 222	Receiver	External Files	fluxmap.csv	
Type 4	Stratified Tank	Tank Volume	1500	m^3
Type 4	Stratified Tank	Fluid specific heat	1.51	kJ/kg-K
Type 4	Stratified Tank	Fluid density	1800	kg/m^3
Type 4	Stratified Tank	Tank loss coefficient	0.1	kJ/hr-m^2-K
Type 4	Stratified Tank	Height of node-[N]	all at 0.4	m
Type 4	Stratified Tank	How many temperature levels should be used in the tank?	20	-
Type 6	Auxiliary Heater	Maximum heating rate	1e12	kJ/hr
Type 6	Auxiliary Heater	Specific heat of fluid	1.51	kJ/kg-K
Type 6	Auxiliary Heater	Overall loss coefficient..	0.1	kJ/hr-K
Type 6	Auxiliary Heater	Set point temperature	540	C
Type 224	Rankine Cycle	Reference Power	11000	kW
Type 224	Rankine Cycle	Reference efficiency	.3400	-
Type 224	Rankine Cycle	Demand Variable**	11000	kW
Type 225	Plant Control	Start Hour	10	Hr
Type 225	Plant Control	Hybridization mode	0	-
Type 225	Plant Control	End Hour	23	Hr
Type 225	Plant Control	Min temp to load	530	C
Type 225	Plant Control	Max temp to heat source	400	C
Type 225	Plant Control	Maximum flow rate to	600000	kg/hr

		heat source		
TRNSYS input file is "Plant Model_strat tank.dck" **Demand variable indicates desired power produced by Rankine cycle				

The notable characteristics of this plant model are the use of a single stratified storage tank which is modeled using 20 vertical nodes (Refer to Figure 58 above for the descriptive plant diagram). Additionally, the plant controller contains new information required by the stratified tank model. Namely, since the fluid temperature leaving the storage tank at both the hot and cold outlets is prone to more widely varying temperatures, the controller implements constraints on the acceptable range of tank outlet temperatures. In the case that the temperature leaving the top (hot outlet) of the tank for the power cycle is not hot enough, the heat transfer fluid is subject to an increased risk of crystallizing after passing through the power cycle heat exchangers or transfer piping since the temperature drop associated with heat transfer to the load forces the HTF temperature to a lower value than would occur if the HTF were to exit storage at a hotter temperature. Likewise, a heat transfer fluid (HTF) temperature at the bottom (cold outlet) of the tank that is too high may cause undue material stress in the receiver and pumps as heat is added in the receiver. Finally, natural limits on pumping capacity and pressure drop through the receiver constrain the HTF mass flow rate through the receiver. The stratified-tank plant controller limits the mass flow rate through the receiver by partially or fully defocusing the heliostat field and reducing the incident radiative power on the receiver surface. The HTF flow rate is modulated to provide a specified receiver HTF outlet temperature, and the ability to limit the HTF mass flow rate is conducive to maintaining a realistic, physical plant model under conditions where the storage tank becomes "over-charged."

A simulation using the stratified tank model and the inputs summarized in Table 33 over the course of a year at the same Daggett, CA, location produces the following characterizing output for the same 172.00 *GWth-hr* incident power as in the previous analyses:

- The total yearly generated electric power is **36.05 *GWe-hr*** and the total thermal power produced by the receiver is **99.98 *GWth-hr***

- The plant capacity factor is **0.374**
- The total conversion efficiency from solar radiation to electric power is on average for the year is **20.96%**
- The average receiver thermal efficiency (not including reflective losses from the surface or pumping losses) is **92.93 %**
- The total annual required pumping power for the receiver is **0.154** ***GWe-hr***

Instead of mapping the tank volume as an indication of the charge level of the thermal storage, the constant volume stratified tank is better described using the hottest, coldest and average temperatures in the tank. The average temperature of the tank approximates the thermal inventory of the tank. When the average temperature in the tank approaches the cold outlet temperature, the thermal storage is nearly fully discharged, and when the average temperature is near the hot outlet temperature, the opposite is true. Figure 65 shows the average tank temperature for the same selected period of time in mid-June used for the previous analyses on the left axis and the HTF flow rates to the receiver and to the power cycle on the right axis.

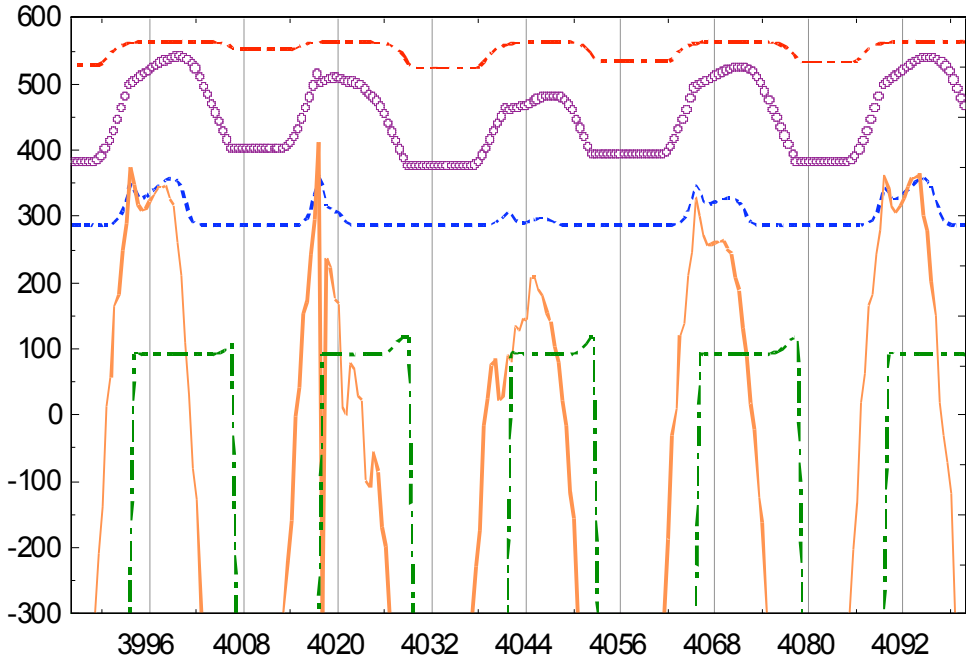


Figure 65: The tank temperature (left axis) and the HTF flow rates (right axis) for the stratified tank plant model for June 14-18th.

Several notable features in Figure 65 emerge that reveal the behavior of the stratified tank thermal storage plant model. The first is that the average tank temperature is strongly affected by the HTF flow rate; at times when the excess flow is coming into the tank to effectively charge the thermal storage (for example at time 4092 in Figure 10) , the cold outlet temperature to the heat source also increases. In the case that the net flow is discharging the tank (such as at time 4029 in Figure 10), the hot outlet temperature to the power cycle decreases. As the tank cold outlet temperature increases, the temperature of the HTF entering the heat source (the receiver or a hybrid backup) increases as well, and this requires that the flow rate to the heat source also increase to maintain the required outlet temperature. Thus for the selected tank size, hot HTF flow into the tank has a compounding negative effect on the system performance since both the cold outlet temperature and the mass flow rate to the heat source increase.

This behavior does not occur in the two-tank system since the thermally charged and thermally discharged HTF is stored in separate tanks. In properly sized systems, the

cold-side tank temperature does not vary with the hot inlet flow, and this behavior demonstrates that either the tank size (1500 m^3) is too small, or the number of nodes used to model the tank (20) is too few. To assess the cause of the behavior, the analysis was repeated with 50 vertical nodes in the stratified tank instead of 20. An additional study was conducted for a smaller number of nodes. The results of these analyses are compared with the results of the 20-node analysis in Table 34.

Table 34: Summary of analysis comparing the effect of additional computational nodes for the stratified tank.

Characteristic	Units	20-node tank	50-node tank	5-node tank
Total annual incident radiation	GW_{th}-hr	172.00	172.00	172.00
Total thermal power	GW_{th}-hr	99.98	100.10	91.99
Total annual generated power	GW_e-hr	36.06	36.15	33.04
Total Receiver pumping power	GW_e-hr	0.154	0.146	0.201
Plant capacity factor	-	0.374	0.375	0.343
Total conversion efficiency	%	20.96	21.01	19.21
Average receiver thermal efficiency	%	92.93	92.94	92.70

The virtually identical results for the 20 and 50-node tank studies strengthen the conclusion that the selected tank size is inadequate to handle the high HTF flow rates experienced during operation. The striking difference between these two result sets and the 5-node results indicate that the number of nodes in the tank does affect the overall model performance and that care must be taken to select a sufficient number of computational nodes for the model. These results also suggest that the total thermal storage capacity for a single stratified storage must be larger than the storage capacity for a two-tank system to avoid destratification.

The second notable feature illustrated in Figure 65 is the dependence of the HTF flow rate to the Rankine cycle on the storage tank hot outlet temperature. As the hot outlet temperature decreases, the HTF flow rate that is required to match the electrical power production in the Rankine cycle increases in order to provide the required energy flow rate. This behavior is seen several times near the end of each daily operation period such as at time 4054 in Figure 65; as the storage hot outlet temperature decreases due to the

net influx of thermally discharged HTF, the mass flow rate to the power cycle ramps up to match the demand.

The final notable model characteristic demonstrated in this plot is the way that the operation is constrained by the hot outlet temperature. Since the average tank temperature indicates the general level of thermal charge left in the storage, it is clear that the plant operation ceases with a considerable amount of charge remaining to be used. In spite of the remaining thermal energy in the tank, the tank hot outlet temperature is confined to be above a specified minimum temperature – in this case 530°*C*. This value is sufficiently high such that the plant shutdown will be triggered with some thermal charge left in the tank. If the limiting value had been set to be closer to the nominal tank cold outlet temperature, the remaining thermal charge in the tank at the end of daily operation would decrease. It should be noted that this concern does not exist with a two-tank system, since the thermally charged fluid remains at isothermal conditions during operation.

The reader may recall that a minimum volume (equated to a minimum thermal charge) level was required for the two-tank plant model. For a stratified tank that is properly sized to match the total inlet and outlet flow rates, the theoretical minimum charge level before the hot-side outlet temperature reaches the minimum value is determined by the vertical position of the thermocline layer in the tank. During thermal discharge of the tank due to a net outflow of hot heat transfer fluid in this tank, the temperature would likely remain approximately equal to the hot inlet temperature until the top of the thermocline layer reached the hot fluid outlet position. However, more knowledge is required concerning molten salt thermocline layers before the minimum charge level can be estimated.

The stratified tank storage plant model also has the ability to run with a supplemental heat source backup like the variable volume model previously discussed (refer to the control scheme presented in Figure 57 and the cycle diagram in Figure 58). The plant

analysis is repeated with the hybridization option enabled. The results for this plant are summarized as follows:

- The total yearly generated electric power is **50.89** ***GWe-hr***
- The total thermal power produced by the receiver is **100.16** ***GWth-hr*** and the total thermal power produced by the fossil-fuel backup is **40.86** ***GWth-hr***
- The fraction of the power produced by the solar source (the solar fraction) is **71.02%**
- The plant capacity factor (or the ratio of the amount of electricity produced compared to the capacity for production) is **0.528**
- The annual Receiver pumping power is **0.140** ***GWe-hr***

The same operation period is plotted to show the tank temperatures and HTF flow rates as a function of time in Figure 66 for the hybrid heat source model.

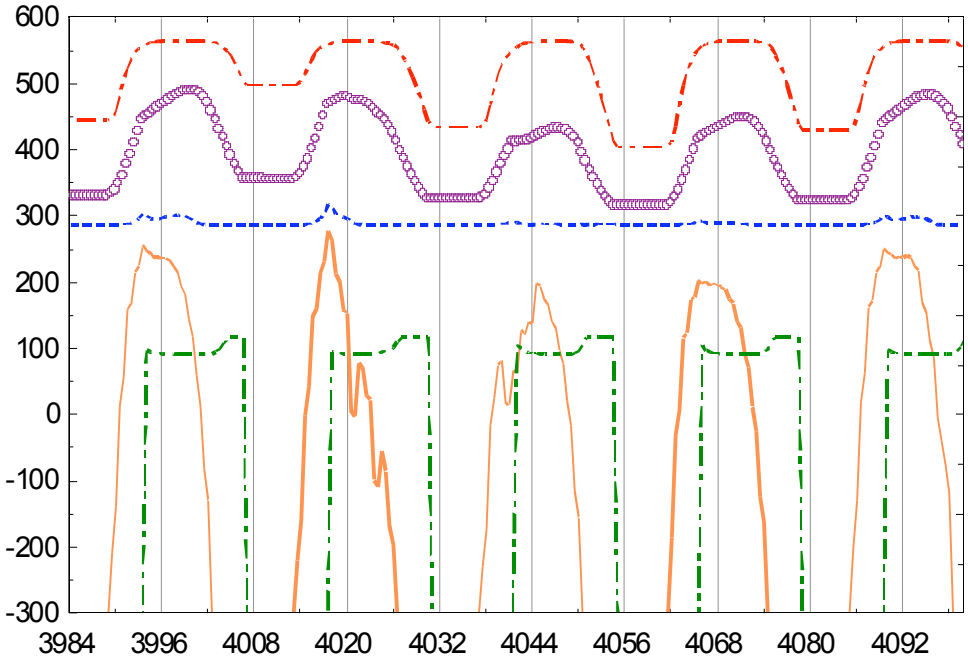


Figure 66: The hot, cold, and average tank temperatures (left axis) and the HTF flow rates (right axis) for the stratified tank storage model with hybrid heat source June 14-18th.

Two major differences exist between the plot presented in Figure 66 and the non-hybrid plot presented previously in Figure 65. The first difference is that the daily cycle operation period extends to the requested end hour of 11:00pm every day due to the available backup heat source capacity. The second difference is in the tank hot outlet temperature. This temperature is no longer limited to be above a minimum specified value; instead, the hot outlet temperature is taken "as-is" and thermal energy is added by the backup heat source to meet the desired Rankine cycle inlet temperature. During times of low solar resource availability, the tank may become completely thermally discharged until energy is available from the receiver to recharge the thermal storage. An example of this scenario is shown in Figure 67, where the hot outlet temperature is reduced to nearly the cold outlet temperature during a period of time in late January. The hybrid heat sources supplements the flow to account for the thermal energy needed to power the Rankine cycle.

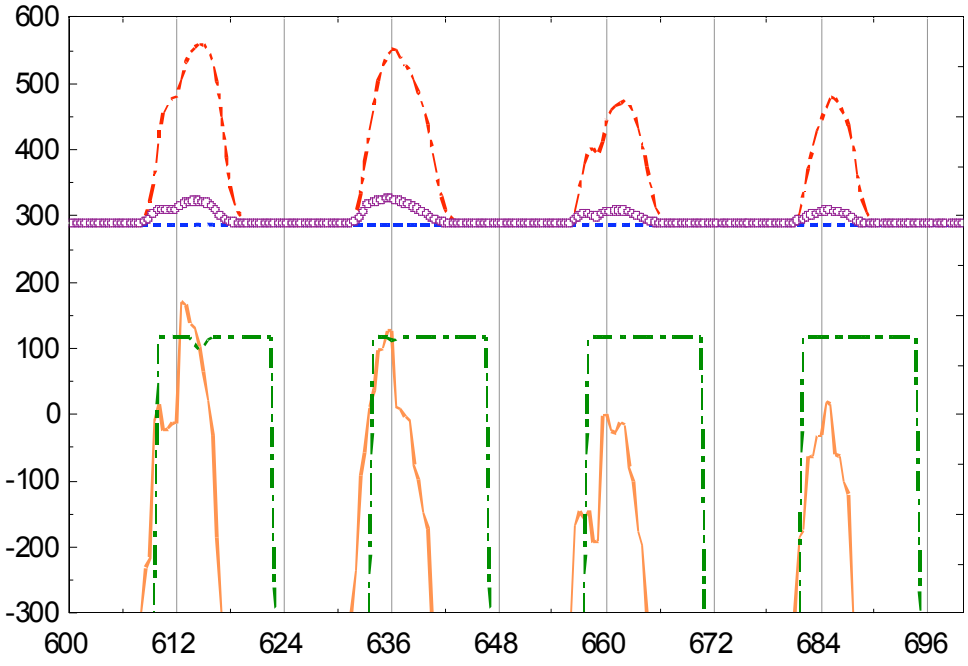


Figure 67: The tank temperature (left axis) and the HTF flow rates (right axis) for a system with a supplemental heat source from January 25-28th.

The previous four analyses are based on the same plant configuration with the exception of thermal storage. Each uses the same receiver flux distribution and heliostat field efficiency as functions of solar position, and each uses the same receiver geometry. Table 35 is provided to consolidate and directly compare the main characteristics observed in each of the previous simulations.

Table 35: A direct comparison of the four plant model analyses performed. "2 tank" corresponds to the two-tank variable volume model, while "1 tank" corresponds to the stratified tank model.

Characteristic	Units	2 tank, no backup	2 tank, backup	1 tank, no backup	1 tank, backup
Total annual incident radiation	GW_{th}-hr	172.001	172.001	172.001	172.00
Total thermal power	GW_{th}-hr	99.46	141.17	99.98	141.02
Total annual generated power	GW_e-hr	36.73	52.20	36.06	50.89
Annual receiver pumping power	GW_e-hr	0.161	0.162	0.154	0.140
Plant capacity factor	-	0.381	0.542	0.374	0.528
Solar fraction	-	1.0	0.707	1.0	0.710
Total conversion efficiency	%	21.36	21.46	20.96	21.01
Average receiver thermal efficiency	%	93.14	93.15	92.93	92.93

Based on the similar results of these analyses, no clear winner emerges between the variable-volume and stratified storage tank. One notable conclusion of this study is that for the modeled system and location, a solar plant with a modest storage capacity and an auxiliary heat source (possibly fossil-fuel) backup can produce a regular and predictable level of power between 10:00am and 11:00pm daily over the course of the year with a solar fraction of approximately 71%. While the total mechanical power (before entering the generator) is slightly higher on average for the two-tank model, the advantage in power production is not decisive. A comparison of the plant total conversion efficiency is also of value and the 2-tank model fares slightly better using this metric. However, the required receiver pumping power and other parasitic operations are not included in this value. The superior conversion efficiency is related to the thermal efficiency of the receiver system, since the stratified tank storage system outputs a higher temperature HTF to the receiver, leading to larger thermal losses from the receiver over the course of operation. However, the difference is nearly trivial in overall plant performance. Depending on implementation issues, subsidy structure, and the circumstances

surrounding the particular application in question, either of the analyzed systems may be the best fit.

6.3.3 Flow Pattern Sensitivity Study

With an understanding of the plant model as implemented in TRNSYS, one particularly illustrative exercise is the analysis of the sensitivity of various outputs to the HTF flow configuration. Discussion in the *Central Receiver Model* chapter (Chapter 3) presented the concept that the receiver HTF flow pattern may be applied as a combination of series and parallel flows. Generally, the HTF can be selected to enter in any panel or number of panels, flow in series through any number of panels, and exit in similar fashion.

Barring unnecessarily complex configurations, an intuitive and logical set of flow configurations was chosen to provide the TRNSYS user with the ability to adapt the flow conditions to the field shape and plant location. The *Flow Pattern* parameter described in the TRNSYS central receiver component (Type 222) contains eight possible selections that each correspond to a distinct configuration. These configurations are for the general case of *N* receiver panels, and have been programmed into the component code. Each arrangement appears in

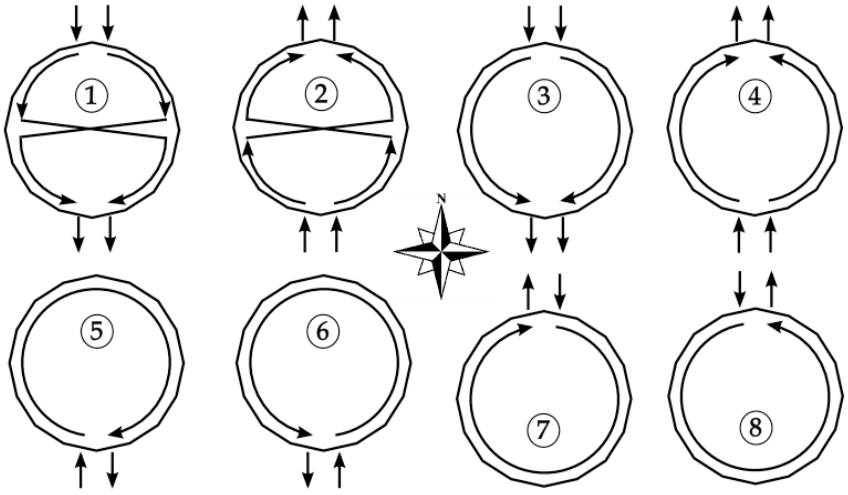


Figure 68.

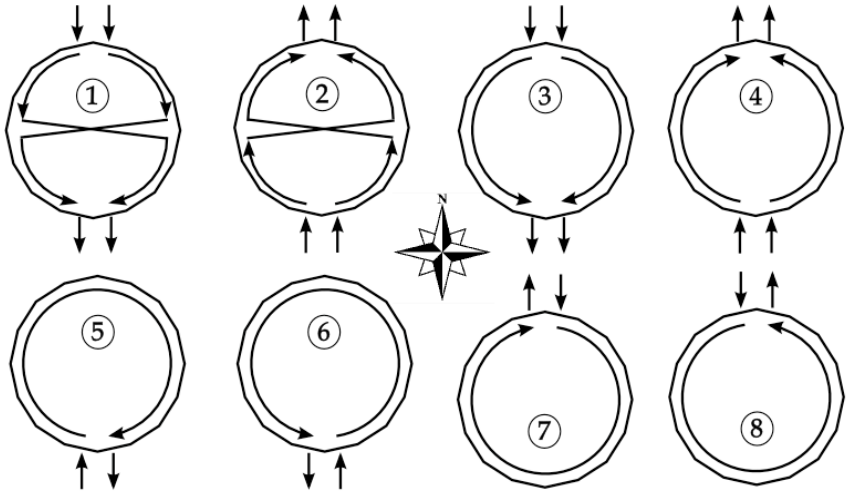


Figure 68: Available coolant flow configurations for the central tower receiver. The coolant enters the receiver in one or two locations, as depicted by the arrows, and follows the path shown to the exit.

An intuitive understanding of the importance of the flow configuration can be achieved by relating the thermal loss experienced by the receiver to the number of panels that sustain high temperatures throughout the day. The receiver is modeled such that the thermal losses are evaluated panel-by-panel around the circumference of receiver based on the convection and radiation coefficients and the surface temperature for each individual panel. For a field that depends heavily on heliostats to the north of the tower, the majority of the reflected solar flux incident on the receiver will likewise be on the north side of the tower. Therefore, a flow configuration that provides cold salt to the north panels will see the major temperature increase occur in the first few panels on the north side of the tower. The flow that travels in series to the south-most panels to exit will have provided a hot heat transfer fluid to the east, west, and south panels which in turn contribute only marginally to the temperature increase. The losses sustained by the receiver in this case will be unnecessarily high. Conversely, the flow pattern that exits towards the North in this situation will have passed through most of the receiver at a much lower temperature, and convection and radiation losses are allayed.

This point is illustrated with the following study: the 11*MWe* tower system that was optimized and designed using the *PTGen* program for the analyses in the previous

subsection was used to determine plant performance for the various available flow configurations. The plant geometry is summarized in Table 36.

Table 36: Design conditions for the sensitivity analysis using the plant developed for previous analysis.

Item	Value
Design Power	11 [MWe]
Individual heliostat size	39.6 [m^2]
Total number of heliostats	1583
Tower height	76.2 [m]
Receiver height	6.2 [m]
Receiver diameter	5.1 [m]
Number of panels	24

The two-tank variable volume storage plant model was used with no supplemental heat source in place. Each of the flow conditions was simulated for one year in the Daggett, CA, location, and the output data were analyzed to evaluate the plant characteristics. The data are summarized in Table 37. For each run, the total annual incident radiation and the total face-plate power (the amount of electrical power generation possible if the plant ran at full nominal capacity for the entire year) are consistent at 172.00*GWth-hr* and 96.37 *GWe-hr* respectively.

Table 37: Summary of the "flow configuration" sensitivity study.

Characteristic	Units	Flow Configuration							
		1	2	3	4	5	6	7	8
Total thermal power	GW_{th}-hr	99.46	100.77	99.46	100.77	100.86	100.86	100.95	100.96
Total annual generated power	GW_e-hr	36.73	37.08	36.73	37.07	37.11	37.11	37.15	37.15
Total receiver pump power	GW_e-hr	0.161	0.164	0.161	0.164	1.011	1.011	1.013	1.012
Plant capacity factor	-	0.381	0.385	0.381	0.385	0.385	0.385	0.385	0.385
Total conversion efficiency	%	21.36	21.56	21.35	21.55	21.57	21.57	21.60	21.60
Average receiver thermal efficiency	%	93.14	94.94	93.14	94.93	94.04	94.03	94.18	94.20

The average receiver thermal efficiency and the average receiver system efficiency (including radiative reflection and pumping losses) for each of the flow configurations are shown in Figure 69.

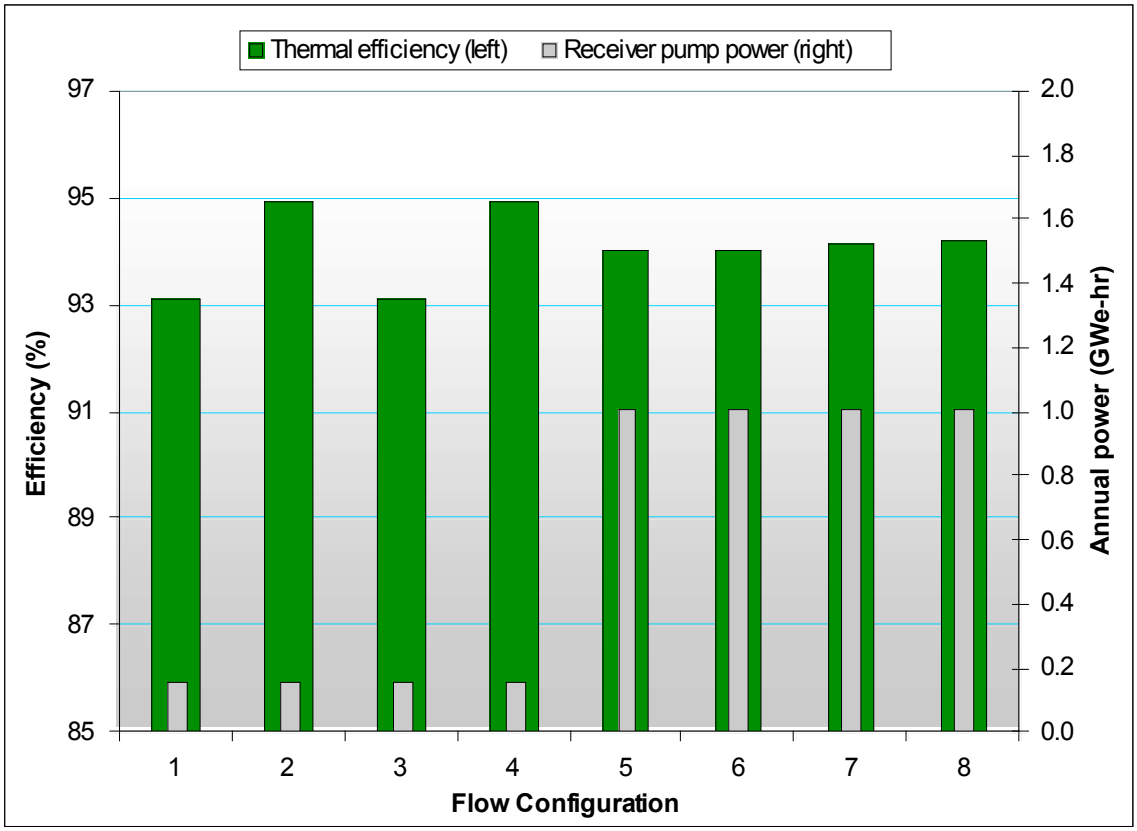


Figure 69: The annual average efficiency of the central tower receiver for different flow configurations and the associated required receiver pumping power.

While the results of this study indicate that a north-based field should be paired with a north-exiting flow configuration, practical concerns may arise that outweigh these findings. The intention to reduce thermal gradients over the length of the receiver panels might be one such situation where a north-to-south flow pattern could alleviate the thermal stress. Of additional concern is the pumping power required to overcome the pressure drop for each configuration. The pumping power shown on the right axis of Figure 69 is significantly higher for flow configurations where all panels are in series (configurations 5-8) than for flow configurations where the panels are arranged in two parallel paths.

The plant models covered in this chapter highlight the versatility of the TRNSYS environment and the component models that have resulted from this research. Despite the models' apparent strengths, some areas of this research would be well served by future investigation and testing. Recommendations in this regard and general conclusions are provided in the following and final chapter.

7 Conclusions and Recommendations

The purpose of this research was to develop a robust fundamentals-based model for the central receiver system power plant for use in long-term transient simulation. The work presented in this report represents fulfillment of that goal, in that a plant optimization methodology was prepared and the tools for long-term detailed performance calculations were developed and implemented in TRNSYS. The major contributions of this research are the development of the PTGen interface and program that makes use of the previously developed DELSOL3 code, the development and implementation of specific plant components in TRNSYS, including the heliostat field (Type 221), central receiver (Type 222), controller for a plant with two-tank storage (Type 223), Rankine power cycle (Type 224), and the controller for a plant with stratified-tank storage (Type 225).

7.1 Recommendations for Future Work

Through the course of this project, several items have emerged as likely candidates for future work. These include resources that have been used in this work but provide insufficient supporting research or are intended as engineering approximations. The most notable examples are improving the receiver external convection correlations developed by Siebers and Kraabel presented in Chapter 3, and the updating the economic and costing models used by DELSOL3. Other candidates for further investigation include work that was initiated or partially developed as part of this research but not pursued to completion due to time limitations. Examples are the cavity receiver model, cavity model convection correlations, radiation view-factor calculation algorithm, and heat transfer fluid property information. These items are also discussed in more detail below.

7.1.1 Cavity Receiver Thermal Model

The development of the cavity central receiver for CRS plants as an alternative to the external cylinder receiver warranted its inclusion in DELSOL3, and the receiver has

continued to find application in modern designs. The most notable application of the cavity receiver is in the PS10 and soon-to-be-completed PS20 CRS plants in Sanlúcar la Mayor, Spain. The cavity model is based heavily on the external cylindrical receiver model that was previously constructed for this research, with additional effort put into the radiation exchange calculations and different convection correlations to accommodate the distinct geometry.

The convection correlations for large, external receiver-type geometries do not apply for the cavity receiver. To implement more accurate correlations, a literature review was conducted. Several papers have been published on the topic of cavity receiver convection losses, including Siebers & Kraabel (1984), Clausing (1981, 1983), and Ma (1993), among others.

This literature is in general agreement on the importance of natural convection in cavity receivers, but somewhat in disagreement on the effects of forced convection. No correlations are available for prediction forced or mixed convection from cavity receivers (Ma, 1993). Studies conducted by Clausing (1981) and McMordie (1984) indicate that low to moderate wind speeds will have a negligible effect on receiver heat loss, while Kugath et al. (1979) and Faust et al. (1981) showed significantly increased thermal losses at the same wind velocities. These studies also indicated that the losses were highly dependent on wind direction and cavity orientation (Ma, 1993).

Although it is difficult to draw any conclusion about the thermal losses from cavity receivers due to forced convection from this literature, one may conclude generally that wind seems to have noticeable effects in small cavity receivers – such as those used in parabolic dish solar collectors – but little effect in larger cavity receivers (Ma, 1993).

The most directly applicable correlation for natural convective loss from a cavity receiver is presented in Siebers and Kraabel (1984), and was later updated by Stine & McDonald (1989). This correlation is shown in Eq. (7.1) below.

$$\overline{Nu_L} = 0.088 \cdot Gr_L^{1/3} \left(\frac{T_w}{T_o} \right)^{0.18} \cdot \cos(\theta)^{2.47} \left(\frac{d}{L} \right)^{s} \quad (7.1)$$

where: s = 1.12 – 0.982*(d/L)

d = aperture diameter

L = receiver internal diameter at cylindrical region

θ = receiver tilt angle

As a first approximation, Siebers and Kraabel (1984) suggest modeling forced convection losses as that of a flat plate, with the mixed convection losses being simply the addition of both forced and natural convection. However, this suggestion is made as an "engineering judgment," and lacks experimental validation.

Radiation exchange takes on more importance in the cavity receiver, since radiation can be exchanged between receiver surfaces, as opposed to the external receiver, where all radiation occurs between a single surface and the surroundings. To adequately characterize this behavior, accurate radiation view factors are required. (A radiation view factor is the fraction of the total radiation emitted from Surface A that strikes Surface B, and is commonly referred to in nomenclature as $F_{A,B}$.) Since no standard view factor library includes analytical solutions for the geometry encountered in the cavity receiver, effort was put into developing an algorithm to numerically calculate surface-to-surface view factors. This effort is discussed in the following section.

7.1.2 Radiation View Factor Calculation Algorithm

The radiation view factor calculation algorithm that was developed is a Monte-Carlo type ray-tracing technique that calculates a vector leaving the originating surface at a random location, angle and elevation, and checks to see if it intersects the polygon on the target surface. Both the originating and target polygons may be any planar shape, with any number of points, and may be convex, concave, or contain holes. The surfaces may lie in any plane in three dimensions.

The algorithm works by employing a fairly simple intersection checking method. The code checks to find the minimum and maximum values of the polygon in the *X*, *Y*, and *Z* dimensions based on the user-defined polygon. This is shown in Figure 70, where the rectangular test domain is defined by the overall minimum and maximum *X*, *Y*, and *Z* values associated with the polygon.

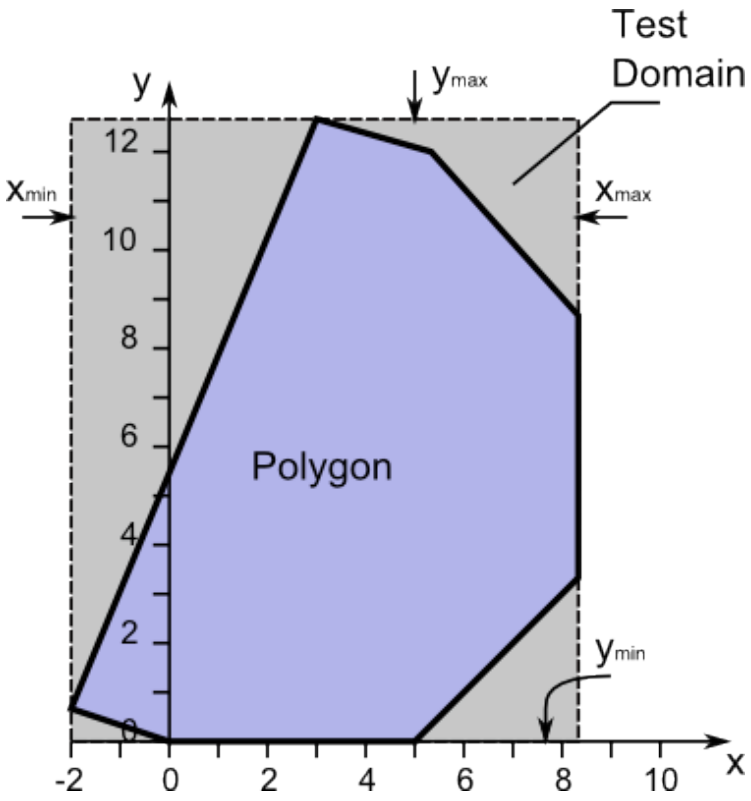


Figure 70: An illustration of the test domain selection for the radiation view factor calculation algorithm.

With these minimum and maximum values, the equation of the source plane is calculated using the first three points specified (thus restricting the polygon to lie in a single plane), and a random point on the plane lying between the minimum and maximum dimensions is chosen. The code then calls an algorithm to determine whether the randomly chosen source point lies in the source polygon. If not, the point is discarded, and a new point is selected. This is done until a valid vector is emitted into space. Figure 71 shows the Y-Z position of 10,000 randomly generated points that lie in the polygon defining the roof of the cavity receiver.

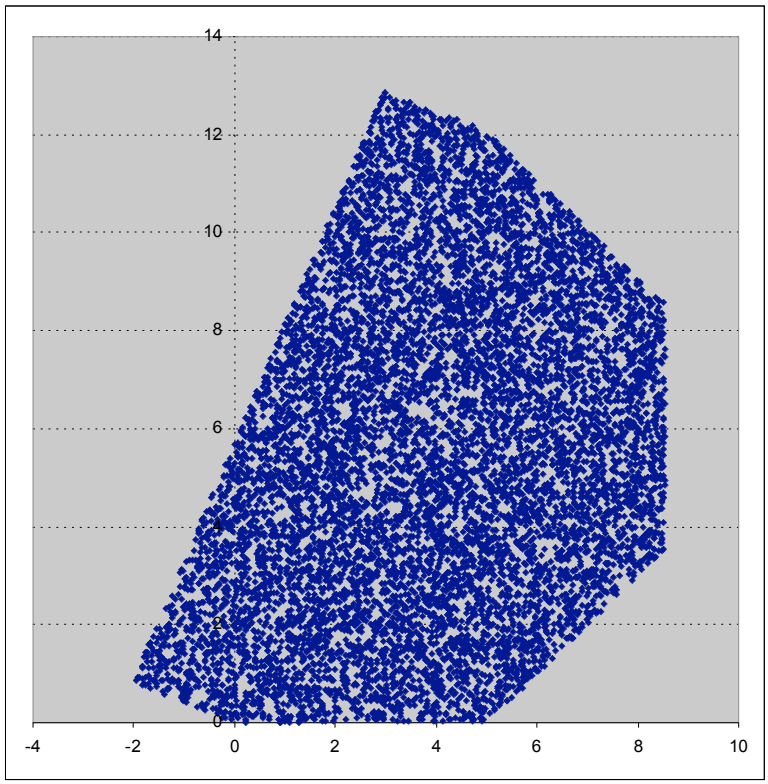


Figure 71: Position of randomly selected points emitting test vectors. Source surface is the top of the cavity receiver, with the longest face of the polygon representing the front of the receiver.

Once the vector is defined, the equation of the target plane is calculated, and the intersection point between the vector and the target plane is determined. In the same way that the source points were checked for inclusion, the intersection point and the target polygon is also tested for inclusion.

The input file for this algorithm includes any number of surfaces defined by any number of coplanar points in the format "(x1,y1,z1),(x2,y2,z2)…" or "[x1,y1,z1],[x2,y2,z2]…". An additional point outside of the plane of the polygon must be specified to indicate which side of the plane is receiving/emitting radiation. This is defined as "{x,y,z}". The code can handle any type of comment or format, so long as the polygon is defined completely on a single line and the direction point is also on that same line. The points

should be ordered in a clock-wise or counter-clockwise manner to avoid polygon edge intersection.

The code will calculate the view factor between each polygon specified in the input file. For example, if four polygons are specified, the algorithm will calculate a total of 4x4 (16) view factors, and arrange them in a table in ".csv" format. Initial tests show good agreement between this algorithm and library view factors. However, additional validation is needed to compare known or analytically calculated view factors with the results obtained from this tool. The algorithm code is included in the digital supplement to this report.

7.1.3 Remaining Items

Several remaining items that should be addressed in future work might be easily overlooked as a detailed model is developed, but have a significant impact on the accuracy of the system model. These include the convection correlations used for both forced and natural convection, and the mixed convection coefficient selected for merging these two loss mechanisms. A survey of applicable literature on convection from large-geometry objects in cross flow yields less-than-conclusive results. The Siebers and Kraabel (1984) study is the most directly applicable for the model in this research since it was conducted to assess the central receiver convective losses. However, the recommendations in that study indicate that certain areas, especially the natural convection coefficient and the mixed convection exponent *m* require additional investigation. Recall that mixed convection is defined such that:

$$h_{mixed} = \left(h_{natural}^{m} + h_{forced}^{m} \right)^{\frac{1}{m}} \tag{7.2}$$

A second area certain to benefit from additional consideration is the heat transfer fluid property information. The HTF property information used in this research was obtained from several reports obtained through collaboration with researchers in the nuclear engineering group at the University of Wisconsin – Madison (Sridharan, 2007). These

molten salts were not used extensively for this research but were made available through correlation in TRNSYS and with the property function lookup in EES. The particular molten salt that was used extensively for this research is 60% $NaNO_3$ + 40% KNO_3, and properties for this material were obtained from Forsberg, et. al (2007). Apparent discrepancy between the property information from Forsberg (2007) and the calculated specific heat capacity in the predictive model provided by Lippke (1995) was found for the specific heat capacity on the order of 5-7% at cycle temperatures. Further investigation into HTF properties is warranted as more advanced fluids are developed.

Also crucial to evaluation of the viability of the technology is a thorough economic and costing analysis to determine the actual capital, operation, and maintenance costs associated with the plant. Further development of the costing parameters and system economics is expected as the model presented in this research is integrated into the Solar Advisor Model.

Appendices

Appendix A: TRNSYS Plant Models

This appendix demonstrates the use of the TRNSYS plant models presented in section 6.3 by summarizing the inputs, outputs, and parameters for the plant models. Both the 2-tank variable volume plant model and the 1-tank stratified storage model will be discussed, since these are the two resulting plant configurations. The TRNSYS input files (*.dck) and project files (*.tpf) are included in the digital supplement described in Appendix D below.

The Two-Tank Storage Plant Model

The TRNSYS two-tank storage plant model takes advantage of the separation of the hot and cold heat transfer fluid (HTF) into two tanks. The total HTF volume is sufficient to fill a single tank; however, the amount of HTF volume in one of the two tanks may range anywhere between the specified maximum and minimum volume level. The total HTF volume is the sum of the HTF volume in each of the individual tanks at any time, and this total volume remains constant throughout plant operation. For more information on the variable-volume plant model, see section 6.1 and section 6.3.1.

The plant model in TRNSYS consist of several components that all serve a unique purpose in the operation of the plant. The interaction between these components also defines the behavior of the plant, and thus, the plant model must be set up appropriately. The component interactions in TRNSYS are defined by connecting outputs from one component to the inputs of one or more other components. For example, in the case of the power tower plant model, the output HTF temperature from the tower receiver is of use for the hot storage tank. The output from the tower component that supplies the HTF temperature can be connected to the input on the hot storage tank that requests the fluid

temperature entering the tank. For the two-tank storage model, the output/input connections are defined in Table 38, and the destination component of the connection is listed in parentheses. Note that in some cases, an output can be used as an input multiple times in several components and will appear more than once in the output list. For example, the solar azimuth angle is used by both the heliostat field and the receiver components, and so a connection from the weather component to each input is reflected in this table. Recall that the components numbered 221-227 were written for this research and other types are standard TRNSYS library components.

Table 38: A list of the TRNSYS model component connections for the variable-volume tank model.

Connected From (output)	→	Connected To (input)
Weather Data (Type 15)		
Dry bulb temperature	→	Environment temperature (HotTank)
Dry bulb temperature	→	Environment temperature (ColdTank)
Dry bulb temperature	→	Dry Bulb temperature (Receiver)
Dew point temperature	→	Dewpoint temperature (Receiver)
Wind velocity	→	wind speed (Heliostats)
Wind velocity	→	Wind velocity (Receiver)
Wind velocity	→	Right axis variable (Graph 3)
Atmospheric pressure	→	Ambient pressure (Receiver)
Direct normal radiation (not interpolated)	→	Direct Normal Radiation (Receiver)
Direct normal radiation (not interpolated)	→	Left axis variable (Graph 3)
Solar zenith angle	→	Solar zenith angle (Heliostats)
Solar zenith angle	→	Solar zenith angle (Plant control)
Solar zenith angle	→	Zenith (Receiver)
Solar azimuth angle	→	Solar azimuth angle (Heliostats)
Solar azimuth angle	→	Azimuth (Receiver)
Hour of the day	→	Hour of the day (Plant control)
Plant Control (Type 223)		
Flow to power cycle	→	Flow rate to load (HotTank)
Flow to power cycle	→	Heat transfer fluid mass flow rate (Rankine Cycle)
Flow to power cycle	→	Right axis variable-2 (Graph 5)

Flow to heat source	→	Flow rate to load (ColdTank)
Flow to heat source	→	Left axis variable (Graph 4)
Field Control	→	Defocus factor (Heliostats)
Supplemental heat source control signal	→	Control Function (Fossil Fuel)
Cycle operation signal	→	Standby control (Rankine Cycle)

Hot Tank (Type 39)

Fluid temperature	→	Heat source inlet temperature (Rankine Cycle)
Fluid temperature	→	Storage hot outlet temp (Plant control)
Fluid volume	→	Actual hot tank HTF volume (Plant control)
Fluid volume	→	Left axis variable-1 (Graph 1)
Fluid volume	→	Left axis variable-1 (Graph 5)
Level indicator	→	Hot Tank Volume Level (Plant control)

Cold Tank (Type 39)

Fluid temperature	→	Storage cold outlet temp (Plant control)
Fluid temperature	→	Fluid inlet temperature (Splitter)
Load flow rate	→	Power cycle requested flow rate (Splitter)
Fluid volume	→	Left axis variable-2 (Graph 1)
Fluid volume	→	Left axis variable-2 (Graph 5)
Level indicator	→	Cold Tank Volume Level (Plant control)

Heliostats (Type 221)

concentrator field efficiency	→	Field eff (Receiver)

Receiver (Type 222)

Salt flow rate	→	Flow from heat source (Plant control)
Salt flow rate	→	Tower flow rate (Splitter)
Salt flow rate	→	Flow from receiver (Mixer)
Salt flow rate	→	Right axis variable-1 (Graph 5)
Receiver thermal eff	→	Left axis variable (Graph 2)
Pump power	→	Input to be printed-3 (System_Printer)
Convection losses	→	Right axis variable-1 (Graph 2)
Radiation losses	→	Right axis variable-2 (Graph 2)
Thermal Power	→	Right axis variable-3 (Graph 2)
Thermal Power	→	Input to be printed-2 (System_Printer)
T_salt_hot	→	Temp from heat source (Plant control)
T_salt_hot	→	Temp from receiver (Mixer)

Rankine Cycle (Type 224)

Cycle power output	→	Right axis variable (Graph 1)

Cycle power output	→	Right axis variable (Graph 4)
Heat transfer fluid outlet temp	→	Inlet temperature (ColdTank)
Heat transfer fluid demand flow rate	→	Load flow demand (Plant control)
Heat transfer fluid mass flow rate	→	Inlet flow rate (ColdTank)
Calculated reference HTF flow rate	→	Reference HTF Mass flow rate (Plant control)
Calculated reference cooling water flow rate	→	Cooling water mass flow rate (Rankine Cycle)
Fossil Fuel (Type 6)		
Outlet fluid temperature	→	Temp from Auxiliary (Mixer)
Outlet fluid flow rate	→	Flow from Auxiliary (Mixer)
Rate of energy delivery to fluid stream	→	Input to be printed-1 (System_Printer)
Splitter (Type 226)		
Fluid outlet temperature	→	Inlet fluid temp (Receiver)
Fluid outlet temperature	→	Inlet fluid temperature (Fossil Fuel)
Flow to heat source	→	Fluid mass flow rate (Fossil Fuel)
Mixer (Type 227)		
Total flow rate	→	Inlet flow rate (HotTank)
Fluid temperature	→	Inlet temperature (HotTank)

The remaining inputs and outputs for the components listed in the table are not connected.

A number of inputs and parameters also require specification in order for the plant model to reflect the behavior of the system under simulation. The following table provides the values used (if not the default value) for the parameter or input for each component in the two-tank plant model.

Table 39: The settings used for the two-tank analysis.

Role	Item	Value	Units
Weather Data (Type 15)			
External Files	Which file contains the TMY-2 weather data?	US-CA-Dagget-23161.TM2	-
Heliostats (Type 221)			
External Files	File name of efficiency file	eff_array.dat	-
Receiver (Type 222)			
Parameter	Number of panels	24	-
Parameter	Receiver diameter	5.1	m
Parameter	Panel Height	6.2	m
Parameter	Tower height	76.2	m
Parameter	Tube outer diameter	21	mm

Parameter	Coolant	12	-
Parameter	Flow Pattern	1	-
Input	Pump efficiency	0.85	-
Hot Tank (Type 39)			
Parameter	Tank operation mode	1	-
Parameter	Overall tank volume	1000	m^3
Parameter	Minimum fluid volume	100	m^3
Parameter	Maximum fluid volume	900	m^3
Parameter	Tank circumference	36.44	m
Parameter	Cross-sectional area	105.7	m^2
Parameter	Wetted loss coefficient	0.1	kJ/hr.m^2.K
Parameter	Dry loss coefficient	0.1	kJ/hr.m^2.K
Parameter	Fluid specific heat	1.545	kJ/kg.K
Parameter	Fluid density	1700	kg/m^3
Parameter	Initial fluid temperature	565	C
Parameter	Initial fluid volume	500	m^3
Cold Tank (Type 39)			
Parameter	Tank operation mode	1	-
Parameter	Overall tank volume	1000	m^3
Parameter	Minimum fluid volume	100	m^3
Parameter	Maximum fluid volume	900	m^3
Parameter	Tank circumference	36.44	m
Parameter	Cross-sectional area	105.7	m^2
Parameter	Wetted loss coefficient	0.1	kJ/hr.m^2.K
Parameter	Dry loss coefficient	0.1	kJ/hr.m^2.K
Parameter	Fluid specific heat	1.545	kJ/kg.K
Parameter	Fluid density	1700	kg/m^3
Parameter	Initial fluid temperature	290	C
Parameter	Initial fluid volume	500	m^3
Rankine Cycle (Type 224)			
Parameter	Reference Power	11000	kW
Parameter	Reference efficiency	.3400	-
Input	Demand Variable	11000	kW
Plant Control (Type 223)			
Input	Start Hour	10	Hr
Input	Hybridization mode	0	-
Input	End Hour	23	Hr
Fossil Fuel (Type 9)			
Parameter	Maximum Heating Rate	$1x10^9$	kJ/hr
Parameter	Specific heat of fluid	1.50	kJ/kg-K
Input	Set point temperature	560	C

In the analyses provided in Chapter 6, several variations were exercised for this model. The variations included investigating the effect of the receiver flow pattern on the overall system performance and investigating the effect of employing a supplemental heat source through fossil-fuel hybridization. These variations used the same connections and inputs listed above with the exception of the applicable parameters in Table 39. The flow pattern was switched using the "Flow pattern" parameter under the *Receiver (Type 222)*

header, while the hybridization option was enabled by setting the "Hybridization mode" input equal to 1 under the *Plant Control (Type 223)* header.

The Stratified Tank Storage Plant Model

The second type of plant model analyzed for this research was the stratified tank storage model. This plant configuration required a single large storage tank designed to facilitate the thermal stratification of the heat transfer fluid. More information on this model is presented in section 6.2 and 6.1.2 of this paper. The information for this plant model is presented in a manner analogous to the previous two-tank model.

Table 40: The TRNSYS component connections for the stratified tank storage model.

Connection From (output)	→	Connection To (input)
Weather Data (Type 15)		
Dry bulb temperature	→	Environment temperature (Storage Tank)
Dry bulb temperature	→	Dry Bulb temperature (External Receiver)
Dew point temperature	→	Dewpoint temperature (External Receiver)
Wind velocity	→	Wind speed (Heliostat Field)
Wind velocity	→	Wind velocity (External Receiver)
Wind velocity	→	Input to be printed-8 (System_Printer)
Atmospheric pressure	→	Ambient pressure (External Receiver)
Direct normal radiation (not interpolated)	→	Input to be printed-9 (System_Printer)
Direct normal radiation (not interpolated)	→	Direct Normal Radiation (External Receiver)
Solar zenith angle	→	Solar zenith angle (Heliostat Field)
Solar zenith angle	→	Zenith (External Receiver)
Solar zenith angle	→	Solar zenith angle (Plant Control)
Solar azimuth angle	→	Solar azimuth angle (Heliostat Field)
Solar azimuth angle	→	Azimuth (External Receiver)
Hour of the day	→	Hour of the day (Plant Control)
Hour of the day	→	Hour of the day (External Receiver)
Storage Tank (Type 4)		

Temperature to heat source	→	Storage cold outlet temp (Plant Control)
Temperature to heat source	→	Inlet fluid temp (External Receiver)
Temperature to heat source	→	Left axis variable-1 (Type65c)
Flowrate to heat source	→	Right axis variable-1 (Type65c)
Temperature to load	→	Storage hot outlet temp (Plant Control)
Temperature to load	→	Inlet fluid temperature (Fossil Fuel)
Temperature to load	→	Left axis variable-2 (Type65c)
Flowrate to load	→	Right axis variable-2 (Type65c)
Average tank temperature	→	Left axis variable-3 (Type65c)

Plant Control (Type 225)

Flow to power cycle	→	Heat transfer fluid mass flow rate (Rankine Cycle)
Flow to power cycle	→	Fluid mass flow rate (Fossil Fuel)
Flow to heat source	→	Hot-side flowrate (Storage Tank)
Field Control	→	Defocus factor (Heliostat Field)
Supplemental heat source control signal	→	Control Function (Fossil Fuel)
Cycle operation signal	→	Standby control (Rankine Cycle)

Heliostats (Type 221)

concentrator field efficiency	→	Field eff (Receiver)

Receiver (Type 222)

Salt flow rate	→	Flow from heat source (Plant Control)
Salt flow rate	→	Input to be printed-5 (System_Printer)
Receiver thermal eff	→	Input to be printed-1 (System_Printer)
Pump power	→	Input to be printed-2 (System_Printer)
Convection losses	→	Input to be printed-3 (System_Printer)
Radiation losses	→	Input to be printed-4 (System_Printer)
Thermal Power	→	Input to be printed-10 (System_Printer)
T_salt_hot	→	Hot-side temperature (Storage Tank)
T_salt_hot	→	Temp from heat source (Plant Control)

Rankine Cycle (Type 224)

Cycle power output	→	Input to be printed-7 (System_Printer)
Heat transfer fluid outlet temp	→	Cold-side temperature (Storage Tank)
Heat transfer fluid demand flow rate	→	Load flow demand (Plant Control)
Heat transfer fluid mass flow rate	→	Cold-side flowrate (Storage Tank)
Calculated reference cooling water flow rate	→	Cooling water mass flow rate (Rankine Cycle)

Fossil Fuel (Type 6)

Outlet fluid temperature	→	Heat source inlet temperature (Rankine Cycle)
Rate of energy delivery to fluid stream	→	Input to be printed-6 (System_Printer)

The remaining inputs and outputs for the components listed in the table are not connected.

Table 41: The settings used for the stratified tank analysis.

Role	Item	Value	Units
Weather Data (Type 15)			
External Files	Which file contains the TMY-2 weather data?	US-CA-Dagget-23161.TM2	-
Heliostats (Type 221)			
External Files	File name of efficiency file	eff_array.dat	-
Receiver (Type 222)			
Parameter	Number of panels	24	-
Parameter	Receiver diameter	5.1	m
Parameter	Panel Height	6.2	m
Parameter	Tower height	76.2	m
Parameter	Tube outer diameter	21	mm
Parameter	Coolant	12	-
Parameter	Flow Pattern	1	-
Input	Pump efficiency	0.85	-
Storage Tank (Type 4)			
Cycle	How many temperature levels (nodes) should be used in the tank?	20	-
Parameter	Tank volume	1500	m^3
Parameter	Fluid specific heat	1.51	kJ/kg-K
Parameter	Fluid density	1800	kg/m^3
Parameter	Tank loss coefficient	0.1	kJ/hr.m^2.K
Parameter	Height of node - *(all nodes)*	0.4	m
Parameter	Maximum heating rate of element *(1 and 2)*	0	kJ/hr
Parameter	Boiling point	2000	C
Derivative	Initial temperature of node -1	560	C
Derivative	Initial temperature of node -2	560	C
Derivative	Initial temperature of node – *(remainder of nodes)*	425	C
Rankine Cycle (Type 224)			
Parameter	Reference Power	11000	kW
Parameter	Reference efficiency	.3400	-
Input	Demand Variable	11000	kW
Plant Control (Type 223)			
Input	Start Hour	10	Hr
Input	Hybridization mode	0	-
Input	End Hour	23	Hr
Fossil Fuel (Type 9)			
Parameter	Maximum Heating Rate	$1x10^9$	kJ/hr
Parameter	Specific heat of fluid	1.50	kJ/kg-K
Input	Set point temperature	560	C

The TRNSYS (.dck) input files can be found on the digital supplement accompanying this report, as detailed in Appendix D below.

Appendix B: Solar II Rankine Cycle Diagram

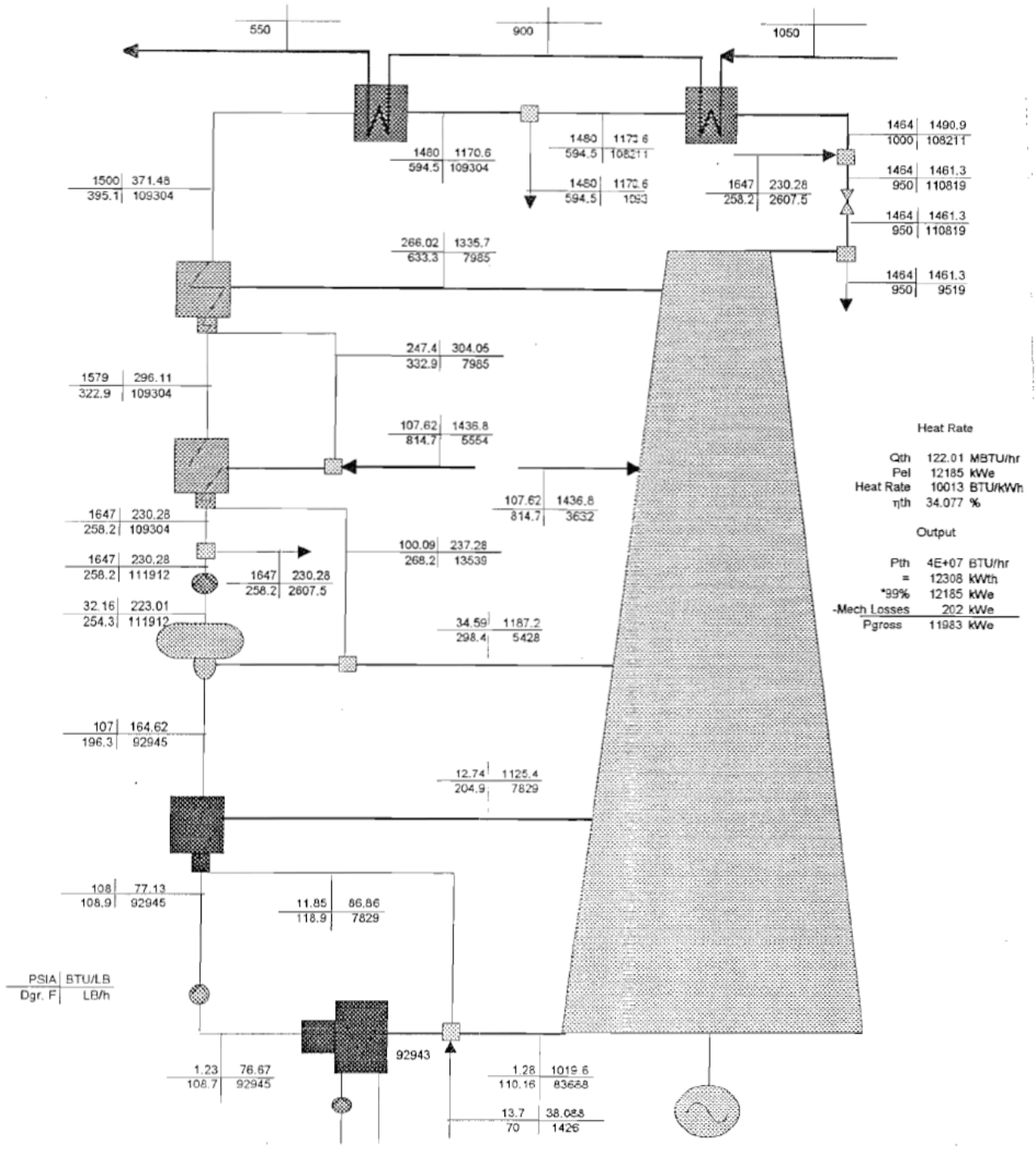
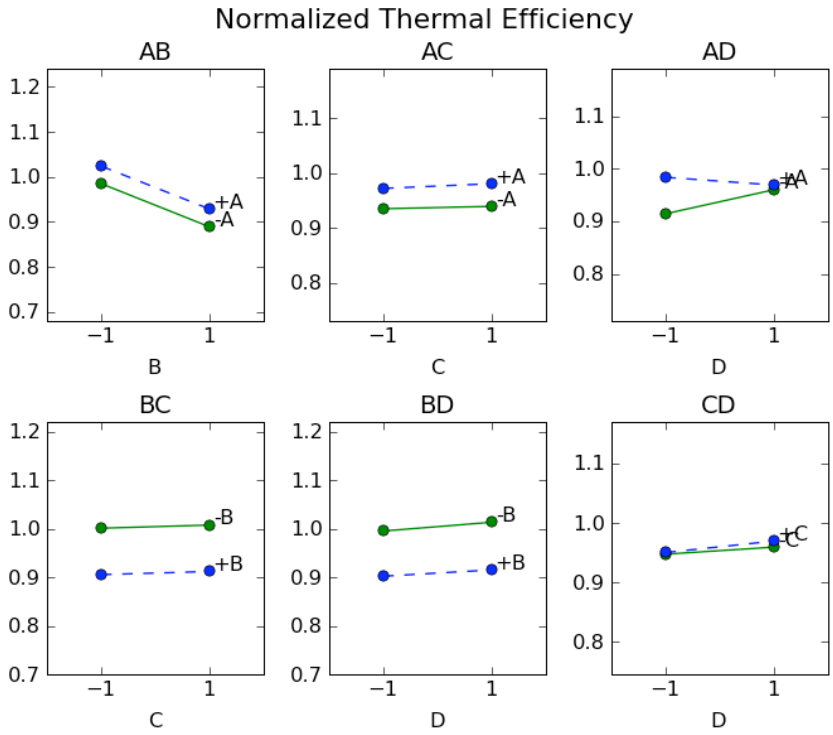


Figure 72: The design Rankine cycle for Solar II, as presented by Lippke, 1995. This cycle was used as a resource in developing the Rankine cycle model for this paper.

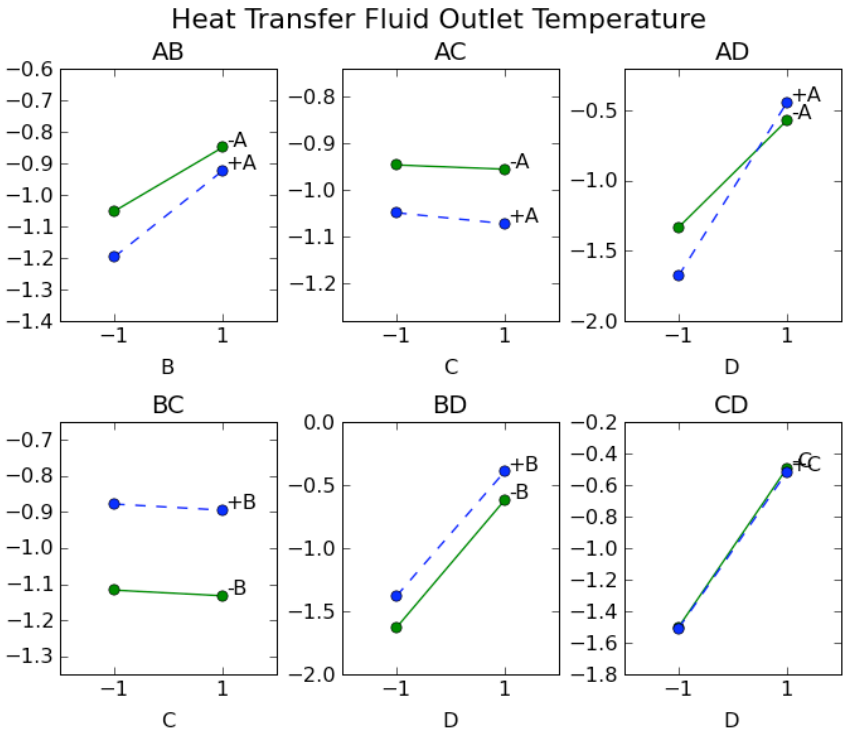
Appendix C: Regression Analysis Interaction Effects

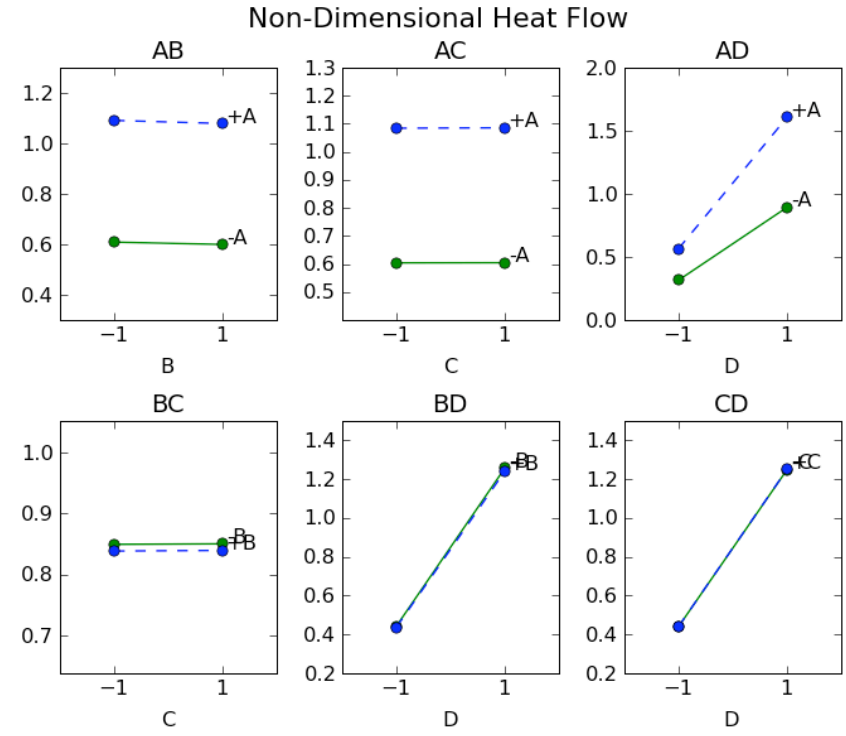
This appendix includes interaction plots for the remaining response variables followed by the Python[1] code written to perform the analysis and create the plots. The plots contain the non-dimensionalized values for the respective response. Interaction is shown in plots where the lines are non-parallel. The variables are assigned corresponding to the following convention:

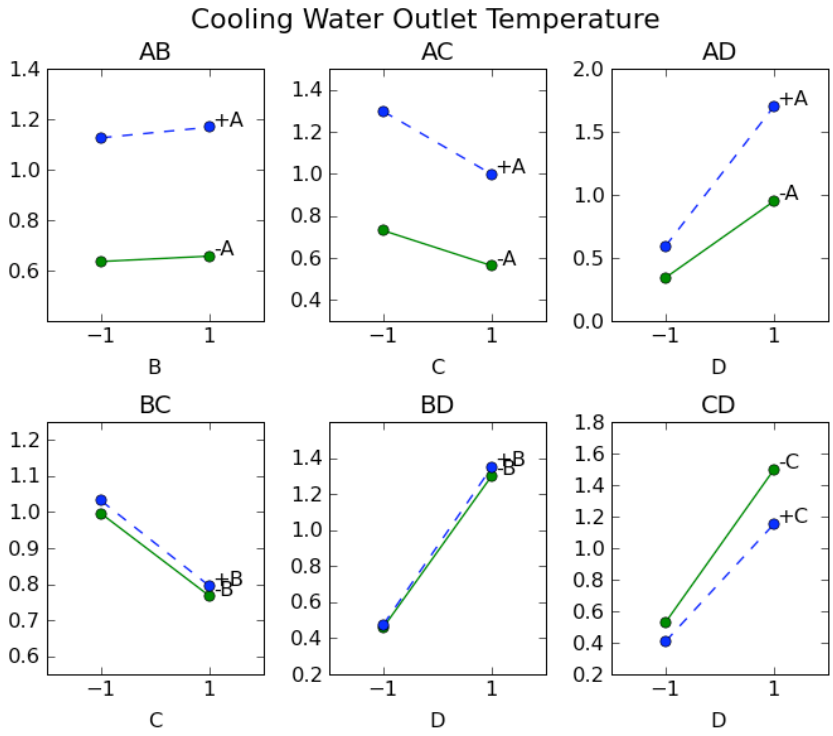
- ***A***: Rankine cycle heat transfer fluid hot inlet temperature
- ***B***: Condenser cooling water inlet temperature
- ***C***: Condenser cooling water mass flow rate
- ***D***: Rankine cycle heat transfer fluid mass flow rate



[1] For more information on the Python programming language, see: www.python.org/doc


Heat Transfer Fluid Outlet Temperature
AB
AC
AD
BC
BD
CD
-A
+A
+B
-B
+C
-C
B
C
D


Non-Dimensional Heat Flow
AB
AC
AD
BC
BD
CD
-A
+A
+B
-B
+C
-C
B
C
D



The normalized values indicated in the previous figures are defined in Chapter 4, but are reproduced here. The normalized efficiency is defined in terms of the cycle efficiency under the given conditions and the cycle efficiency at design conditions.

$$\eta_{ND} = \frac{\eta_{cycle}}{\eta_{design}} \quad [1]$$

The heat transfer fluid outlet temperature from the power cycle is defined relative to the temperature of the steam at the boiler pressure, since the driving temperature difference in the cycle is represented by the hot HTF inlet temperature minus the steam temperature at boiler pressure. This temperature difference is scaled by the same temperature difference, but at cycle design conditions.

$$T_{htf,cold,ND} = \frac{T_{htf,cold} - T_{sat,ref}}{T_{sat,ref} - T_{htf,cold,ref}} \quad [2]$$

The normalized heat flow from the HTF to the steam within the cycle is simply defined by the ratio of the heat flow at given conditions to the heat flow at design conditions.

$$\dot{q}_{heat,ND} = \frac{\dot{q}_{heat}}{\dot{q}_{heat,ref}} \quad [3]$$

Finally, the normalized cooling water outlet temperature is defined. This is also defined in terms of the cycle design value.

$$T_{cw,ND} = \frac{T_{cw,out} - T_{cw,in}}{\Delta T_{cw,ref}}$$
$$where: \quad [4]$$
$$\Delta T_{cw,ref} = T_{cw,out,design} - T_{cw,in,design}$$

The following code provides a listing for the statistical calculation methodology used to determine the significant effects and variable interactions described in Chapter 4. An electronic version of this Python-language code is included in the digital supplement, described below.

```
"""This script calculates the main effects and interaction effects
for a 2^4 experiment.  The data is loaded from an array called
'studydata.csv', which should includes a column for each variable to be
evaluated.  Each row in the column corresponds to a test where factors
A, B, C, and D are evaluated at hi/lo levels ([+,+,+,+] is the first level set,
followed by [+,+,+,-], [+,+,-,+] and so on).
Statistical reference: Wu & Hamada (2000). John Wiley and Sons, Inc. """
print 'Importing libraries...'
import numpy
import csv

A=[] ; B=[] ; C=[] ; D=[]
data=[] ; var=[]

for i in range(1,17):
    if i<9: A.append(-1)
    else: A.append(1)
    if (i in range(1,5)) or (i in range(9,13)): B.append(-1)
    else: B.append(1)
    if i in list((1,2,5,6,9,10,13,14)): C.append(-1)
    else: C.append(1)
    if i%2. != 0.: D.append(-1)
    else: D.append(1)
```

```
var.append([A,B,C,D]) ; var=var[0]  #append the effects ; simplify the array
print 'Reading source file...'
#open file
file=open('studydata.csv')
csvfile=csv.reader(file)
num_resp = [len(line) for line in csvfile][0]
file.seek(0,0)

for i in range(num_resp):  data.append([]) #setup data array
#load data
for line in csvfile:
    j=0
    for entry in line:
        data[j].append(float(entry))
        j=j+1

print 'Calculaing effects...'
#calculate main effects for A,B,C,D
main_effects=[]
for column in data:     #responses
    effects=[]
    for variable in var:  #A,B,C,D
        i=0 ; pls=[] ; mns=[]
        for point in column:
            if variable[i]==1:  pls.append(column[i])
            else:  mns.append(column[i])
            i=i+1
        #now calculate effect
        effects.append(sum(pls)/float(len(pls))-sum(mns)/float(len(mns)))
    #add to main effects
    main_effects.append(effects)

#Calculate 1st degree interactions
int_set = [[0,1],[0,2],[0,3],[1,2],[1,3],[2,3],
           [0,1,2],[0,1,3],[0,2,3],[1,2,3],[0,1,2,3]]
int_effect=[] ; effect_sort=[] ; PSE = [] ; k=0 ; t_stat=[] ; all_effect=[]
for column in data:
    effects=[]
    for group in int_set:
        pls=[] ; mns=[] ; i=0
        for point in column:
            #--check the sign of the group ([q,p,r]-> sign(q*p*r))
            count=0 ; negative=False
            for element in group:
                if var[element][i]<0: count=count+1  #count negative values
            if count%2. != 0: negative=True  #if count is odd, sign is negative
            #--
            if negative:  mns.append(point)
            else:  pls.append(point)
            i=i+1
        #calculate effect
        effects.append((sum(pls)/float(len(pls))-sum(mns)/float(len(mns)))/2.)
        effect_sort.append([pls,mns])   #keep track of the sorted pluses and
                                        #minuses for later
    #add variable effects to interaction effect array
    int_effect.append(effects)


    #Now, use Lenth's method (p132 Wu & Hamada, 2000) to calculate the PSE
    #.. (pseudo standard error.) This can be used to determine the t-statistic
    #..to test significance.  t = theta/PSE
    all_effects = [abs(i) for i in main_effects[k]]+[
        abs(j) for j in int_effect[k]]
```

```
    s_0 = 1.5*numpy.median(all_effects)
    mod_effects=[]
    for theta in all_effects:
        if abs(theta) < 2.5*s_0:
            mod_effects.append(abs(theta))
    if mod_effects:
        PSE.append(1.5*numpy.median(mod_effects))
    else:
        PSE.append(0)

    #And calculate the t-statistic, using 1% significance, which corresponds to
    #.. 3.63 for 15 effects (2^4-1)
    if PSE[k]!=0:
        t_stat.append([i/PSE[k] for i in all_effects])
    else:
        t_stat.append([0 for i in all_effects])
    all_effect.append(all_effects)
    k=k+1


print 'Constructing output file...'
#Construct an output file
outfile = open('effects.csv','w')
#write main effects
outfile.write('Effects\n ,A,B,C,D,AB,AC,AD,BC,BD,CD,ABC,ABD,ACD,BCD,ABCD,\n')
i=0
for variable in all_effect:
    outfile.write('var'+str(i+1)+',')
    for node in variable:  outfile.write(str(node)+' , ')
    outfile.write('\n')
    i=i+1
#write t-statistics
outfile.write('t_statistics (Lenth\'s method)\n')
i=0
for variable in t_stat:
    outfile.write('var'+str(i+1)+',')
    for node in variable:  outfile.write(str(node)+' , ')
    outfile.write(','+str(PSE[i])+',')
    outfile.write('\n')
    i+=1

outfile.close()

#------------------------------------------------------------
#create interaction plots
print 'Importing plotting tools...'
import matplotlib
matplotlib.use('WXAgg')
from pylab import arange, sin, pi
from matplotlib.backends.backend_wxagg import FigureCanvasWxAgg as FigureCanvas
from matplotlib.figure import Figure
from wx import *

class CanvasFrame(Frame):
    def __init__(self):
        Frame.__init__(self,None,-1,
                       '2-Variable Interaction Plots for Regression Analysis',
                       size=(750,650))

        #We need to make 6 plots for the interaction effects, and this needs
        #.. to be done for each of the response variables.  First, create the
        #.. tabs in the window.
```

```
        #create the general notebook framework
        self.notebook_1 = wx.Notebook(self, -1, style=0)
        self.notebook_1_panes=[] ; i=0 ; self.sizer = []    #zeros
        for column in data:
            #Create an array of vertical boxsizers of length (#response vars)
            self.sizer.append(BoxSizer(VERTICAL))
            #create number of tabs in the notebook of length (#response vars)
            self.notebook_1_panes.append(Panel(self.notebook_1, -1))
        #
        #Sort the data so that we can create the 6 plots
        #.. Sort criteria will be:
            #sort by variable 1, variable 2
            #..(+) and (-)
            #1st subscript)  Response variable
            #2nd subscript)  Plot (1-6)
        int_set2 = int_set[0:6] #only use the [x,y] groups
        responses=[]    #zeros
        for column in data:  #each column is a response variable
            points=[]
            for group in int_set2:  #For each of the two-factor interactions
                hihi=[] ; hilo=[] ; lohi=[] ; lolo=[] ; i=0     #zeros
                #look at each point for the groups in the interaction, and sort
                #..them into 4 groups
                for point in column:
                    if var[group[0]][i]<0 and var[group[1]][i]<0:
                        lolo.append(point)
                    if var[group[0]][i]<0 and var[group[1]][i]>0:
                        lohi.append(point)
                    if var[group[0]][i]>0 and var[group[1]][i]<0:
                        hilo.append(point)
                    if var[group[0]][i]>0 and var[group[1]][i]>0:
                        hihi.append(point)
                    i=i+1
                #Calculate the 2 lines (2 points per line)
                hihiave=sum(hihi)/float(len(hihi))
                hiloave=sum(hilo)/float(len(hilo))
                lohiave=sum(lohi)/float(len(lohi))
                loloave=sum(lolo)/float(len(lolo))
                points.append([loloave,lohiave,hiloave,hihiave])
            responses.append(points)

        Response_titles = ["Non-Dimensional Power",
                           "Non-Dimensional Heat Flow",
                           "Heat Transfer Fluid Outlet Temperature",
                           "Normalized Thermal Efficiency",
                           "Cooling Water Outlet Temperature"]
        for i in range(len(self.notebook_1_panes)):
            #add the canvas to the sizer.. the canvas contains all 6 plots
            self.sizer[i].Add(self.plots(responses[i],self.notebook_1_panes[i],
                                         Response_titles[i]),1, LEFT|TOP|GROW)
            self.notebook_1_panes[i].SetSizer(self.sizer[i])   #Set the sizer
            #Add the pane to the tab

self.notebook_1.AddPage(self.notebook_1_panes[i],Response_titles[i])
        self.sizer_main = BoxSizer(VERTICAL)
        self.sizer_main.Add(self.notebook_1,1,EXPAND,0)
        self.SetSizer(self.sizer_main)

    def plots(self,response,parent,filename):
        """this creates a series of 2x3 plots,
        each with 2 data series plotted on them"""
        plotnames=['AB','AC','AD','BC','BD','CD']
        self.SetBackgroundColour(NamedColor("WHITE")) #set the background color
```

```
        self.figure = Figure()  #Create the figure
        self.axes=[] ; i=0 ; x=[-1,1]   #zeros
        for plotdat in response:    #for each of the plots
            self.axes.append(
                self.figure.add_subplot(2,3,(i+1))) #Add the individual subplot
            self.axes[i].plot(x,plotdat[0:2],'go-')  #plot the data for series1
            self.axes[i].plot(x,plotdat[2:4],'bo--')  #plot series 2
            self.axes[i].set_title(plotnames[i])    #set the subplot title
            self.axes[i].set_xlim([-2,2])           #set the x axis limits
            self.axes[i].set_xlabel('\n'+plotnames[i][1])    #set the x label
            self.axes[i].text(
                1.1,plotdat[1],'-'+plotnames[i][0])   #set the series names
            self.axes[i].text(1.1,plotdat[3],'+'+plotnames[i][0])   #same
            y=self.axes[i].get_ylim()       #check the y-axis limits
            self.axes[i].set_ylim([y[0]-.2,y[1]+.2])  #adjust the y-axis limits
            self.axes[i].set_xticks([-1,1]) #format the x-axis tick marks
            i=i+1
        self.figure.suptitle(filename,fontsize=16)
        self.canvas = FigureCanvas(parent, -1, self.figure)
        #Set the space between the subplots
        self.figure.subplots_adjust(wspace=.3,hspace=.4)
        self.figure.savefig(filename+'.png')
        return self.canvas

    def OnPaint(self, event):
        """When the window is resized, redraw the image"""
        self.canvas.draw()

print 'Generating plots...'

class App(App):
    """Define the application"""
    def OnInit(self):
        'Create the main window and insert the custom frame'
        frame = CanvasFrame()
        frame.Show(True)
        return True

#Create an instance of the App, and send it to the MainLoop event handler
app = App(0)
app.MainLoop()
```

Appendix D: Digital Supplement

This appendix contains a listing of the documents and files contained in the digital supplement (CD) to this thesis. The included items consist of programs, scripts, and documents related to the TRNSYS, Fortran, and other areas of development discussed in this research.

The included items are organized into several categories. The first category includes items required to run the PTGen interface and program. Note that all of the programs, subroutines, and functions listed in this appendix were written by the author for use in the research presented in this paper. The exception to this statement is the DELSOL3 code, which was modified to accommodate modern compilers and use in this application. DELSOL3 was originally written by Kistler, et al. (1986) of Sandia National Labs, Albuquerque, NM.

Folder: *PTGen*

PTGen_ees.exe	The GUI interface for PTGen
PTGen.exe	The executable program called by the GUI
macro.emf	The EES macro used by the GUI interface
cities_info.txt	A text resource file containing the formatted names of all of the locations included in the GUI. This file can be modified to include additional locations if needed
Storage.txt	A resource file for the GUI indicating storage and plant oversize suggestions

PTGen.sln	The MS Visual Studio 2005 'solution' file that manages compilation of the program code

The *PTGen* folder contains a subfolder of the same name. This folder holds the Fortran code used to generate the PTGen executable file.

Folder: *PTGen\PTGen*

trunk.f90	The main calling program structure for the PTGen executable. This code manages the operation of the various subroutines – including calling DELSOL3 – to produce the full set of information required for a TRNSYS simulation.
dattodelsol.f90	A subroutine used to convert the PTGen GUI output to a DELSOL3 formatted input file.
opttoazel.f90	A subroutine used to scrape information from the DELSOL3 plant optimization run output and construct the DELSOL3 input file for the heliostat field efficiency calculation as a function of solar azimuth/zenith angle.
opttoflux.f90	A subroutine used to scrape information from the optimization run output and construct a DELSOL3 input file for the receiver flux map calculation runs.
fluxmap.f90	A subroutine that converts the DELSOL3 output from the flux map calculation runs to a format readable for the long-term TRNSYS simulation. The file produced by this subroutine (fluxmap.csv) is used by Type 222.
azeltotrnsys.f90	A subroutine that converts the DELSOL3 output from the solar azimuth/elevation vs. field efficiency calculation to a

format readable by Type 221 in TRNSYS. The file produced by this subroutine is named *eff_array.dat*.

DELSOL3MJW.for — The DELSOL3 Fortran code modified for use in this program. The specific changes and modifications are documented in Chapter 5, and the code is designated as the modified version by the addition of the letters *MJW*, which represent the initials of the author of this paper.

plant_info.f90 — A Fortran subroutine that produces a summary file containing the results of the DELSOL3 plant optimization run. **The information in this file represents only the output produced by DELSOL3, and does not represent the results of any long-term TRNSYS simulation.**

mylen.f90 — A Fortran function that determines the length of a character string, including spaces, but without leading or trailing blanks.

textsearch.f90 — A Fortran function that searches a data/text file for a specific string, and returns the specific row and column where the string was located in the file.

The second major category of items included in the digital supplement is the detailed component models for long-term simulation in TRNSYS. These items include the TRNSYS component files, the component image files, and the Fortran files used to construct the model.

Folder: Components

Type22x.tmf	The TRNSYS proforma file, where *x* represents any of the numbers between 1 and 7.
Type22x.bmp	The image files that accompany the proformas. These icons are used by the Simulation Studio to represent the component.

Within the *Components* folder are six sub-folders, each containing the Fortran files for the components. The seven components each have their own folder with the exception of the plant controller components (Types 223 and 225) which share a folder.

Folder: Components\Control Components

Type223.for	The main Fortran subroutine for the variable-volume storage plant controller component. This subroutine interacts with the TRNSYS program, managing input and output.
ControlLogic.f90	The subroutine called by *Type223.for* that contains the control logic for that type.
Type225.for	The main subroutine for the stratified storage plant controller component.
ControlLogic_b.f90	The subroutine called by *Type225.for* that contains the control logic for that type.

Folder: Components\Heliostat Component

Type221.for	The subroutine containing the heliostat field component. This component requires an external file containing the

heliostat field efficiency as a function of solar position (default eff_array.dat).

Folder: Components\Splitter

Type226.for — The subroutine containing the TRNSYS splitter component. This component splits a single flow into two separate flows, depending on the demanded flow rate from one of the flow destinations.

Folder: Components\Mixer

Type227.for — The subroutine containing the TRNSYS mixer component. This component simply merges two flow streams into one.

Folder: Components\Rankine Component

Type224.for — The main Rankine cycle component subroutine that interacts with the TRNSYS calling program. This subroutine manages the inputs and outputs.

RankineCycle.f90 — The Rankine cycle subroutine that contains the calculations determining the behavior of the Rankine cycle. This subroutine is managed and called by *Type224.for*.

coef_arrays.f90 — A subroutine that contains the coefficients for the curve fits for cycle power and cycle heat addition used in the Rankine cycle regression model. This subroutine is managed by *Type224.for.*

Design Rankine Cycle.ees	The EES program used to design the Rankine cycle heat exchanger sizes and fluid flow rates. This model is discussed in detail in Section 4.1.
Performance Rankine cycle.ees	The EES program used to analyze the design cycle. The heat exchanger sizes and fluid flow rates determined in the design cycle are controlled to mimic the behavior of a physical Rankine cycle. This model is discussed in detail in Section 4.2.
2FWH Design.png	An image file containing a schematic of the Rankine cycle diagram used in the design and performance stages of the analysis.
Regression_analysis.py	The Python-language script used to calculate the statistical effects of the system under consideration. This script is presented in Appendix C above, and is also included here.
studydata.csv	A sample input file used in the regression analysis. The file contains five columns of data, each corresponding to an input effect. These are: Non-Dimensional Power, Non-Dimensional Heat Flow, Heat Transfer Fluid Outlet Temperature, Normalized Thermal Efficiency, and Cooling Water Outlet Temperature, respectively. There are 16 rows of data, each corresponding to the 16 runs in the factorial experiment.
effects.csv	This file contains sample output from the regression analysis script. Each effect is calculated and its significance is indicated with the t-statistic table. For definition of the letter variables (A-D) see Appendix C.

Folder: Components\Tower Component

Type222.for	The receiver subroutine that interacts with the TRNSYS calling program by managing inputs, outputs, and parameters.
CRS-main.f90	The main tower receiver subroutine that interacts with the *Type222.for* subroutine. This Fortran file contains the main receiver calculations to determine the performance of the receiver under various conditions. This subroutine also calls other supplementary subroutines and functions.
fluxinterp.f90	A subroutine called by *CRS-main.f90* that reads the receiver flux distribution file, and selects the best match for the flux map based on the current solar position. Contrary to the terminology suggested by the subroutine name, this subroutine does not interpolate flux maps, but only selects the closest match. For justification of this method, see Chapter 3.
flowPatterns.f90	A subroutine containing the receiver flow pattern calculations required to specify the path that the HTF takes through the receiver.
PipeFlow.f90	A subroutine that implements the Nusselt number correlation for internal fully developed flow in a tube.
Nusselt_FC.f90	A function that implements the Nusselt number correlation for external forced convection from the receiver surface as presented by Siebers and Kraabel (1984).

properties.f90	A function that provides property data for the fluids and materials used in the CRS model.
skytemp.f90	A function that calculates the effective sky temperature given a dew point temperature, ambient temperature, and hour of the day.
convert.f90	A function that converts between two groups of units.
CRS Receiver Model.EES	The EES model used to generate the Fortran version of the receiver model. Note that this model may differ from the Fortran model, and does not exactly represent the final model presented for this research.
Nusselt_FC.txt	An implementation of the function described above for use in the EES model.

To facilitate the use of the previously described component models, two plant models have been developed and included in this supplement. The two-tank and stratified-tank plant models described in Chapter 6 are provided here. Although it is recommended that the user of these plant models develop their own plant configurations using PTGen, the files required for an example plant configuration are provided with these plant models. Note that both plant models use the same flux and field efficiency data files. Finally, two scripts that were used to analyze the annualized results of the long-term TRNSYS simulations are included.

Folder: TRNSYS Plant Models

Plant Model_two tanks.tpf	The TRNSYS plant model file for the two-tank storage configuration presented in Section 6.1 and 6.3.1.

Plant Model_strat tank.tpf	The TRNSYS plant model file for the stratified storage configuration presented in Section 6.2 and 6.3.2.
fluxmap.csv	A sample flux map input file for the 11*MWe* plant presented in Chapter 6. This file is used by the tower component (Type 222).
eff_array.dat	A sample field efficiency file for the 11*MWe* plant presented in Chapter 6. This file is used by the heliostat field component (Type 221).
array_view.csv	A more human-readable version of the information provided by *eff_array.dat*. This file is not used in the simulation by any component, but is provided for review by the user.
US-CA-Daggett-23161.TM2	The weather data file for Daggett, CA, used to provide weather information for the simulation.
plant_summary_2tank.py	A Python-language script that calculates several helpful parameters relating to the plant performance of the 2-tank configuration. This script is written to accommodate only the output provided by the *Plant Model_two tanks.tpf* TRNSYS project simulations.
plant_summary_strat.py	A version of the file described above, but applied to the *Plant Model_strat tank.tpf* project.

The radiation view factor algorithm presented in Chapter 7 is also included in this supplement. A sample geometry definition file and a sample output matrix are supplied.

Folder: View Factor Algorithm

view_factor.py	The script containing the view factor calculation algorithm.
Sample Geometry.txt	A file containing the sample geometry definition method. For more on the method used to define geometry, see Chapter 7.
Sample output.csv	A sample output file demonstrating the array of view factors for the geometry presented in the sample geometry file.

Finally, several miscellaneous items of importance are included in the supplement. Among these are the DELSOL3 user manual (Kistler, 1986), the molten salt property lookup tables implemented in EES, and the Python platform installer file. The file names are respectively:

**Folder: **

DELSOL3 user manual.pdf
Molten salt properties.zip
python-2.6.1.msi

References

DOE, "Basic Research Needs for Electrical Energy Storage", Report of the Basic Energy Sciences Workshop on Electrical Energy Storage, April 2-4, www.sc.doe.gov/bes/reports/files/EES_rpt.pdf, (2007).

Energy Information Agency. Table 5.18 Crude Oil Domestic First Purchase Prices, 1949-2007. Retrieved 12/1/08 from http://www.eia.doe.gov

Forsberg, C.W., Peterson, P.F., Zhao, H. (2007). High-Temperature Liquid-Fluoride-Salt Closed-Brayton-Cycle Solar Power Towers. *Journal of Solar Energy Engineering, 129, 141-146.*

El-Wakil, M.M. (1984). *Powerplant Technology.* McGraw-Hill Book Company.

Energy Department Documents and Publications/ContentWorks. "DOE Funds 15 New Projects to Develop Solar Power Storage and Heat Transfer Projects For Up to $67.6 Million." via COMTEX, September 22, 2008.

Falcone, P.K. (1986). *A handbook for solar central receiver design*. Sandia National Labs., Livermore CA. Technical Report SAND-86-8009

Fox, R.W, McDonald, A.T., Pritchard, P.J. (2006). *Introduction to Fluid Mechanics*. John Wiley and Sons, Inc.

FPL Energy. Retrieved 12/2/2008 from http://www.fplenergy.com/portfolio/contents/segs_viii.shtml

Gnielinski, V. (1976). *New equations for heat and mass transfer in turbulent pipe and channel flow.* International Chemical Engineering. Vol. 16, pp359.

Herrmann, U., Kelly, B., Price, H. (2004). *Two-tank molten salt storage for parabolic trough solar power plants.* Energy, Vol. 29, pp. 883-893.

Hottel, H. C. (1976). *A simple model for estimating the transmittance of direct solar radiation through clear atmospheres*. Solar Energy, Vol. 18, pp. 129.

Image courtesy: http://cache.gettyimages.com

Image courtesy: Image Number SAND-064232d. Sandia National Laboratories Applied Energy Database System. http://www.energylan.sandia.gov/

Karni, J., Kribus, A., Doron, P., Rubin, R. Fiterman, A., Sagie, D. (1997). The DIAPR: A High-Pressure, High-Temperature Solar Receiver. *Journal of Solar Energy Engineering, 119, 74-78.*

Kearney, David W., and Miller, Charles E., (1988). *SEGS VI: Technical Evaluation of Project Feasibility*. Submitted to Luz Solar Partners VI.

Kiera, M., Schiel, W. (1989). *Measurement and Analysis of Heliostat Images*. Journal of Solar Energy Engineering, Vol 111, Issue 1, pages 2-9.

Kistler, B.L. (1986). *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 National Labs, Albuquerque, NM. SAND86-8018.

Kolb, G.J., Diver, R.B., Siegel, N. (2007). *Central-Station Solar Hydrogen Power Plant*. Journal of Solar Energy Engineering. Vol. 129, Issue 2, pages 179-184.

Kribus, A. et. al (2001). *Performance of the Directly-Irradiated Annular Pressurized Receiver (DIAPR) Operating at 20 bar and 1,200°C.* Journal of Solar Energy Engineering (ASME), Vol. 123, p10.

Lata, J.M., Rodriguiz, M., Alvarez de Lara, M. (2006). *High Flux Central Receivers of Molten Salts for the New Generation of Commercial Stand-Alone Solar Power Plants*. SENER publication.

Leary, P.L., Hankins, J.D. (1979). *User's guide for MIRVAL: a computer code for comparing designs of heliostat-receiver optics for central receiver solar power plants*. Sandia National Labs, Livermore, CA. SAND-77-8280.

Lippke, F. (1995). *Solar II Overall Efficiency at Reduced Receiver Outlet Temperatures*. Sandia National Labs., Albuquerque, NM.

Lipps, F.W., Vant Hull, L.L. (1977). *A cellwise method for the optimization of large central receiver systems*. International Solar Energy Society, Annual Meeting, Orlando, Fla., June 6-10, 1977.

Litwin, R.Z. (2002). *Receiver System: Lessons Learned from Solar Two*, SAND2002-0084. Sandia National Laboratories, Albuquerque, NM.

Leary, P.L., Hankins, J.D. (1979). *User's guide for MIRVAL: a computer code for comparing designs of heliostat-receiver optics for central receiver solar power plants.* Sandia National Labs, Livermore, CA. SAND-77-8280.

McMahan, A.C. (2006). *Design and Optimization of Organic Rankine Cycle Solar-Thermal Power Plants.* (Masters Thesis, University of Wisconsin – Madison, 2006).

Moore, J.W., Stanitski, C.L., Jurs, P.C. (2002). Chemistry: The molecular science. Harcourt College Publishers, Orlando FL.

National Renewable Energy Lab CSP Technology Workshop, March 7, 2007. Retrieved 12/3/2008 from http://www.nrel.gov/csp/troughnet/.

National Renewable Energy Lab website information, retrieved from www.nrel.gov 12/1/2008

Nellis, G.F., Klein, S.A. (2009) *Heat Transfer*. Cambridge University Press. Manuscript submitted for publication.

Ortega, J.I., Burgaleta, J.I., Tellez, F.M. (2006). *Central receiver system (CRS) solar power plant using molten salt as heat transfer fluid.* SENER / CIEMAT publication, Madrid, Spain.

Pacheco, J.E., et al. (2002). *Final Test and Evaluation from the Solar Two Project*. Sandia National Labs, Albuquerque, NM. SAND2002-0120.

Patnode, A. (2006). Simulation and Performance Evaluation of Parabolic Trough Solar Power Plants. (Masters Thesis, University of Wisconsin – Madison, 2006).

Personal communication with Dr. Kumar Sridharan (August, 2007). University of Wisconsin – Madison.

Reilly, H.E., Kolb, G.J. (2001). *An Evaluation of Molten-Salt Power Towers Including Results of the Solar Two Project.* (SAND2001-3674), Sandia National Laboratories, Albuquerque, NM.

Romero, M., Buck, R., Pacheco, J. (2002). *An Update on Solar Central Receiver Systems, Projects, and Technologies*. Journal of Solar Energy Engineering, 124(2), 98-108.

Sánchez, M., Romero, M. (2005). *Methodology for generation of heliostat field layout in central receiver systems based on yearly normalized energy surfaces.* Solar Energy, Vol. 80, p861-874.

Schwarzbözl, P. (2006). *A TRNSYS Model Library for Solar Thermal Electric Components (STEC): Reference Manual Release 3.0.* Deutsches Zentrum für Luft und Raumfahrt e.V. (DLR), D-51170 Köln, Germany.

Schwarzbozl, P. (2006). *A TRNSYS Model Library for Solar Thermal Electric Components (STEC)* (Reference Manual, Release No. 3.0). Deutsches Zentrum für Luft und Raumfahrt e.V. (DLR). D-51170 Köln, Germany.

SolarPaces PS10 summary page. Retrieved 12/1/2008 from: http://www.solarpaces.org/Tasks/Task1/PS10.HTM

Stine, W.B, Geyer, M. (2001). *Power From The Sun.* Retrieved from online text from: http://www.powerfromthesun.net/book.htm.

Stodola, A., and Loewenstein, Louis C. (1945). *Steam and Gas Turbines*. Volume I. New York: McGraw-Hill Book Company.

Stone, K.W., Jones, S.A. (1999). *Analysis of Solar Two Heliostat Tracking Error Sources*. Sandia National Laboratory, Albuquerque, NM, and The Boeing Company, Huntington Beach, CA. SAND99-0239C.

Walzel, M. D., Lipps, F. W., and Vant-Hull, L. L. (1977). *A Solar Flux Density Calculation for a Solar Tower Concentrator Using a Two-Dimensional Hermite Function Expansion.* Solar Energy, Vol. 19, pp. 239-253.

Wendelin, T. (1989). *SOLTRACE: A NEW OPTICAL MODELING TOOL FOR CONCENTRATING SOLAR OPTICS.* Solar Engineering 1989, American Society of Mechanical Engineers.

Williams, D.F. (2006). *Assessment of Candidate Molten Salt Coolants for the NGNP/NHI Heat-Transfer Loop*. Oak Ridge National Laboratory, Oak Ridge, Tennessee. ORNL/TM-2006/69