Spring 4-2001

Analysis of Power Generation System Utilizing a Salt Gradient Solar Pond

Charlotte Elizabeth Walker
University of Tennessee-Knoxville

Senior Honors Project:
Analysis of a Power Generation System Utilizing a Salt Gradient Solar Pond

Prepared by:
Charlotte E. Walker, Student
Mechanical, Aerospace and Engineering Science Department

Advised by:
Dr. Robert J. Krane, Professor
Mechanical, Aerospace and Engineering Science Department

Presented to:
Dr. Thomas W. Broadhead, Director
The University Honors Program

The University of Tennessee, Knoxville
Knoxville, TN 37916

University Honors 458
May 8, 2001

Abstract

Salt gradient solar ponds are an environmentally friendly energy resource, capable of providing 60 - 85°C water for process heating, desalination, and energy generation. The possibility of generating power from a salt-gradient solar pond is explored, using a flash separator and the pond's brine as the working fluid for the power cycle. The current design focuses on a 5 MW peaking power plant to be stationed in Bakersfield, California.

The design suggests that the flash separation process will be capable of generating 0.0075 kg steam/kg brine, and will require a 14 m diameter and 42 m long flash drum operating at approximately 28700 Pa. The turbine will need to produce 7 MW of power in order to run the pumps, fans, vacuums associated with the process. The condenser and cooling tower will need to be able to handle 97.6 m^3/min (23,148 gal/min) of water, which is on par with the flow rate of the Tennessee River. The turbine and condenser will also need to be placed 12 m above the surface level of the pond, and the pumps will need to be buried 5 m below the separator. The overall first law efficiency of the power cycle is around 5%, compared with a Carnot efficiency of around 14%. The initial estimate of the pond area suggests that 2,500,000 m^2 (620 acre) should provide enough thermal energy for the pond to operate at peek efficiency for 8 months of the year.

The design analysis suggests that while the system is theoretically possible, it may be physically impractical and economically unsound.

Introduction

The effective and cost efficient utilization of solar energy has long been a goal of the environmental community because solar energy is inexpensive, clean, and renewable. However, utilizing solar energy possesses a unique set of challenges. These include finding a way to concentrate the dilute solar energy and to store this energy, since the source is only available during daylight hours. The salt-gradient solar pond addresses these concerns by providing an inexpensive, large-area solar collector with the potential to store the sun's energy in the lowest layer of the pond. The thermal energy stored in the pond can then be converted to mechanical and electrical energy.

Solar Ponds in El Paso, Texas, and the Dead Sea, Israel, have been successfully used for power generation. However, these facilities have used organic power cycles, using heat exchangers to transfer the energy from the pond to the power cycle's working fluid. However, the heat transfer process is inefficient, necessitating larger ponds. It is possible to generate steam directly from the pond's brine using a flash separation process. This steam can then be used as the working fluid for the pond's power cycle.

Statement of Purpose

The design of the pond's power cycle requires knowledge of the basic thermodynamic principles as well as more specific knowledge of the appropriate implementation of the specific pieces of equipment. The purpose of this project is to combine the general and specific knowledge to develop a 5 MW peeking power cycle for a salt-gradient solar pond that implements a flash separator and the pond brine as the working fluid.

Salt-Gradient Solar Pond

Solar ponds are naturally occurring phenomena that occur in many locations around the world where a salt-gradient (or halocline) is sufficiently steep and the pond is protected from winds that would tend to mix the waters and disturb the thermal gradient. These ponds have been found in locations as remote as Lake Vanda in the Antarctic where the temperature at the bottom of the pond was recorded to be 25°C despite the fact that the top of the pond was frozen and the ambient temperature was -20°C. Medve Lake in Transylvania has also been studied as a naturally occurring solar pond, reaching temperatures as high as 70°C in the summer. Other naturally occurring lakes have been recorded in Washington State and Israel [1].

Artificial ponds have been created both for research and industrial purposes throughout the world. Israel may have the most experience, with the creation of a 5-MW power plant using a solar pond at the Dead Sea. However, the United States, India and Australia have also built and run solar ponds, such as the Miamisburg, OH, location where a 2000-m^2 pond is used to provide heating for a public swimming pool [1]. A research pond in El Paso, TX, has been used to provide process heating, electricity, and fresh water [2].

Artificial salt-gradient solar ponds are typically constructed in three layers of clear salt water, of different salinities and thermal properties, on top of a black bottom. On top is a low-salinity region called the Upper Convective Zone (UCZ), followed by a layer of increasing salinity called the Non-Convective Zone (NCZ). The lowest layer usually has a salt concentration at near saturation levels and is called the Lower Convective Zone (LCZ). A sketch of a solar pond is shown in Figure 1. When the solar radiation contacts the solar pond, the infrared and longer wavelengths are absorbed by UCZ. However, the visible and ultraviolet light passes through the different zones to be absorbed by the non-reflecting bottom of the pond and heats the water in the LCZ. The NCZ serves as an insulating layer that traps the heat at the bottom of the pond [3].

This unusual behavior for the salt-gradient pond is brought about by the increase in density and decrease in thermal conductivity associated with the increase in salinity. The increase in density due to the salinity traps the hotter water at the bottom of the pond by counteracting the decrease in salinity associated with heating the water. The pond must be carefully maintained to prevent boiling of the liquid in the LCZ, which would mix the pond waters and destroy the salt-gradient [4].

The maintenance of a salt-gradient pond requires that salt and water be added to maintain the salt-gradient and to make up for evaporation losses. The pond must also be protected from strong winds, which would tend to mix the waters at the upper levels of the pond, and the growth of algae or other sources of fouling which would cloud the waters of the pond. The pond would also lose some of its efficiency if the bottom of the pond were to be become reflective, as the solar radiation would then be partially reflected out of the pond. Other concerns are the effects of the extraction of the energy itself, which if too extreme might disturb the steady state performance of the pond [4].

The temperature difference that exists between the LCZ and UCZ in solar ponds can be used to drive heat engines and power cycles. Although the overall efficiency is low, due primarily to the low temperature difference (approx. 50°C), the inexpensive nature of the energy makes it an attractive possibility.

Salt-gradient solar ponds are simple in concept, easy to construct, and inexpensive to build. However, they do have a few disadvantages. For instance, leakage of the brine could foul a local freshwater supply. They are also not very efficient, and should be constructed close to the equator so that the solar radiation will be fairly constant throughout the year.

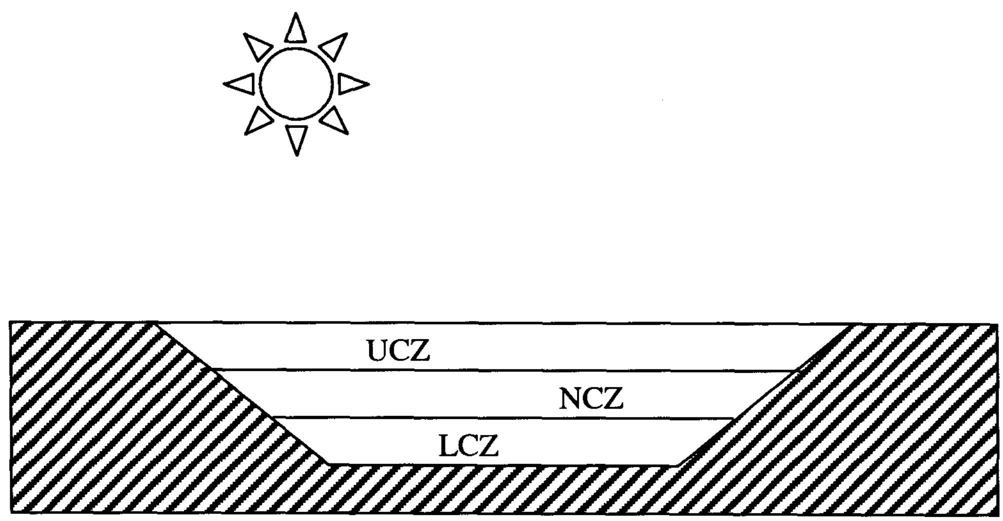


Figure 1: Diagram of a salt-gradient solar pond showing the various layers.

Power Cycles

The salt-gradient solar pond can be used to generate electricity from the thermal difference between the hot and cold layers of the pond by using a power cycle. A cycle is defined in thermodynamics as series of processes that have no net change of state. In other words, although a system may undergo changes of state, if these changes return it to its original state, the system may be said to have undergone a cycle. A power cycle is particular cycle that extracts work from a circulating fluid and transfers it to the surroundings. (As opposed to a refrigeration or heat pump cycle, which transfers energy to a circulating fluid from the surroundings.)

However, to understand the limits of a power cycle, a basic understanding of the first and second laws of thermodynamics is necessary. The first law of thermodynamics states that although the energy in the system may change forms (e. g. from heat to work), the total amount of energy is conserved. The second law states that heat energy is not spontaneously transferred from a cooler body to a warmer body.

Since a cycle can have no net energy change, the net amount of work must equal the net heat transfer of the cycle. The heat transfer is controlled by the absolute temperature difference between the hot and cold layers of the pond, which in turn controls the amount of work (and thus power) that can be extracted from the cycle.

The thermal efficiency of a power cycle is defined as the ratio of work to heat input, which is kept from unity by the heat that is rejected from the cycle. The second law of thermodynamics leads to a concise theoretical maximum for the thermal efficiency of a power cycle, known as the Carnot efficiency:

$$\eta_{\max} = 1 - \frac{T_{cold}}{T_{hot}}.$$

The Carnot efficiency can be applied to a Carnot cycle, which is a theoretical cycle that is operated without losses due to irreversibilities inherent in real physical processes.

A variation on the Carnot cycle that can be applied, at least in part, to the power cycle used in the salt-gradient solar pond is the Rankine cycle. The Rankine cycle consists of 4 proceses, in which the working fluid is

1. expanded in a turbine (allowing work to be extracted as the pressure of the working fluid drops), then
2. passed through a condenser (where heat is removed at constant pressure), then
3. compressed back to the turbine's inlet pressure (which is work done on the fluid), and then
4. flows through a boiler at constant pressure to return the working fluid to its original temperature and state (here, heat is transferred to the fluid.)

In order to get a better understanding of a Rankine cycle, each of the processes will be examined individually in terms of using water as the working fluid.

The turbine is perhaps the most visible of the components of the cycle, since it is here that the power to run the generator is produced. A turbine develops power by passing the high temperature, high pressure working fluid through a series of blades that are attached to a shaft. As the fluid expands through the turbine, the fluid pushes on the blades which causes the shaft to rotate, and this rotation can be used to power generators or other devices. The fluid itself experiences a pressure drop as a result of its interaction with the turbine. The pressure drop within the turbine is a primary design criterion for the system, since as the pressure drops, the working fluid will change phase from a super heated vapor to a two-phase mixture of liquid and vapor (or if the fluid enters the turbine as a saturated vapor, as the pressure drops, the amount of liquid will increase.) The presence of liquid in the working fluid leads to increased stresses on the turbine blades, eventually destroying them. The cycle must be designed to create the maximum pressure drop (thus the maximum power extracted) that will allow for a minimum of excess wear on the turbine itself.

The condenser takes the low-pressure steam or water/steam mixture and condenses it to a saturated liquid. A condenser is a heat exchanger that transfers heat away from the working fluid and induces a phase change from vapor to liquid. Theoretically, this would be a constant pressure process, but frictional losses within the condenser make this impossible, and condensers must be designed to minimize these losses. It should be noted that there are several types of condensers that can be used, depending on the application. Heat exchangers typically operate by inducing heat transfer from a warmer fluid to a cooler fluid. (In the case of a condenser, the fluid to be condensed is the warmer fluid.) However, the two fluids may be arranged to flow in parallel, cross-flow, shell and tube, and counter-current flow patterns. In addition, the heat exchanger may allow for some mixing of either or both fluids.

A Rankine cycle compresses the fluid back to the turbine's inlet pressure using a pump. The selection of a pump is based on the required energy change at the inlet and exit of the pump. For most systems, this is a function of the density of the fluid and the required pressure change; however, if there were a significant change in velocity or potential energy between the inlet and exit, this would also need to be taken into account.

The boiler used to heat the liquid to the steam that enters the turbine is also a heat exchanger, however, the working fluid of the power cycle is the cooler fluid to be heated.

The power cycle that will be used for the salt-gradient solar pond is a variation on a simple Rankine cycle. The pond cycle begins at the solar pond, with the solar energy heating the water instead of a boiler. Hot brine is pumped from the pond into a flash separator, where the fluid is sent through a throttle valve, which drops the pressure of the fluid below that required to send it into the liquid/vapor phase, consisting of a more concentrated brine and a nearly pure water vapor. The hot brine is pumped back to the bottom of the pond, while the water vapor is sent to the turbine, as before. After passing through the turbine and the condenser, the liquid is returned to the top of the pond.

The flash separator is one of the key pieces of equipment in the pond's power cycle. Although it is a proven technology that is relatively simple in concept, it is not very efficient over the small pressure range allowed. The pressure of the brine entering the separator will be primarily a function of the height of the separator relative to the top of the pond, which will be at atmospheric pressure. The flash separator will generate steam from about 1-2% of the brine flowing through it. Considering the amount of low-pressure steam required to generate 5 MW of power, this could lead to unmanageable mass flow rates from the solar pond.

The turbine for the pond's power cycle will be a standard turbine, except that it will have to operate at unusually low pressures for a power cycle, complicating its design and selection. However, a 5 MW power cycle will require a relatively small turbine.

The condenser will be a direct contact heat exchanger, which allows the fluids to mix with each other. The direct contact of the fluids provides a higher efficiency when the purity of working fluid is not a major issue. The condenser will produce essentially pure water which can be returned to the top of the pond to partially offset losses in water due to evaporation. A mechanical vacuum pump will be connected to the condenser to discharge the inert gases that will seep into the sub atmospheric system. The cold water for the condenser will be provided by a cooling tower, which will be fed by the hot water leaving the condenser.

The pond's power cycle will not use a pump to return the fluid to the pond, as might be expected due to the pressure difference between the condenser outlet and the pond surface, but will instead rely on a change of height (and thus potential energy) to provide the necessary energy to raise the pressure back to ambient levels. To do this, the turbine and condenser will have to be raised above the level pond by 50-100 meters (depending on the exit pressure of the condenser).

The pond's power cycle also calls for a cooling tower to provide cold water for the condenser. The hot water leaving the condenser will be cooled in the tower, and part of the cooled water will be transferred back to the pond and part will be pumped back to the condenser. The cooled water may also be used as the sealing fluid for a liquid ring vacuum pump, which will be used to help maintain the vacuum in the condenser.

A schematic of the pond's components is shown in Figure 2.

The pond's power cycle has the potential to generate a modest amount of power at a reasonable cost; however, the tolerances involved in the design require careful attention to details to insure that a reasonably cost effective design is implemented.

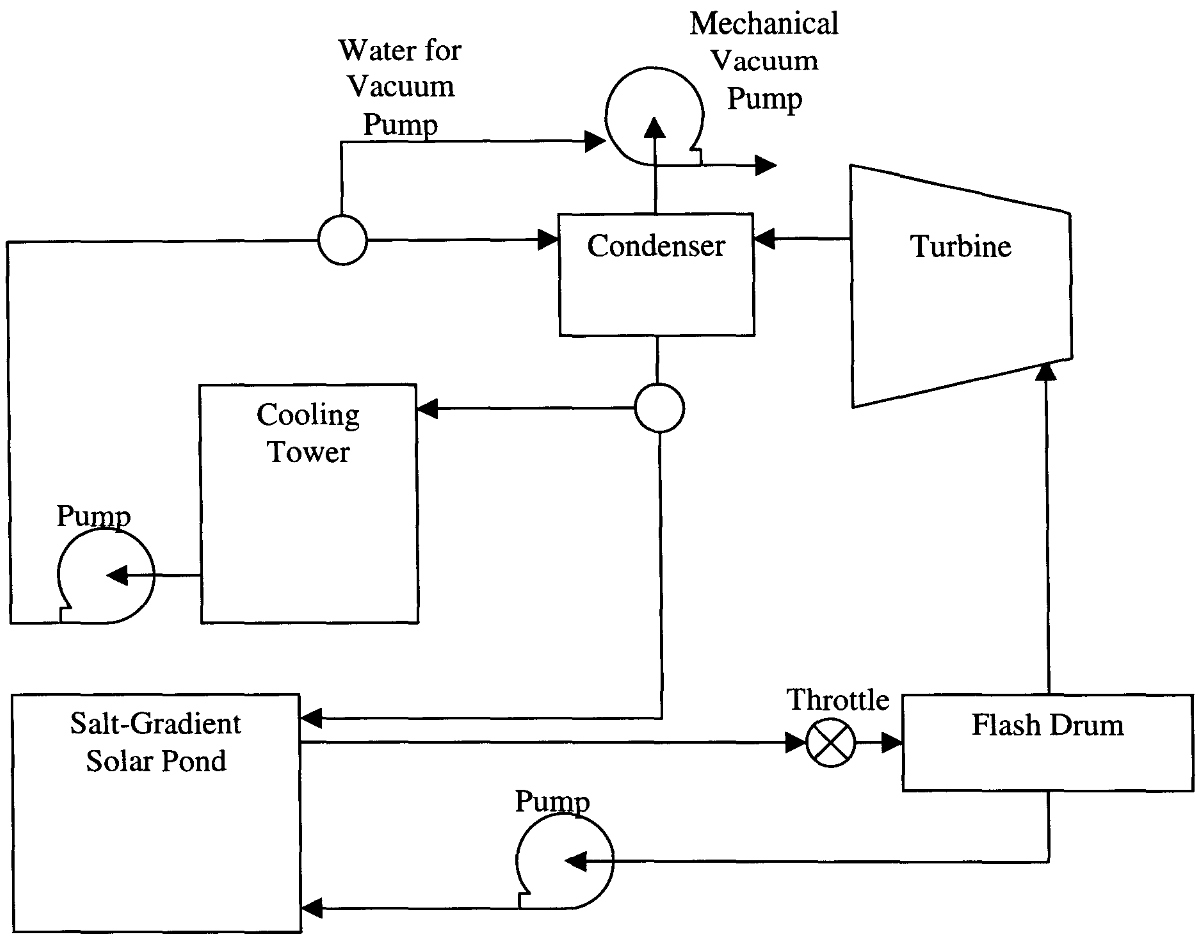


Figure 2: Schematic of salt-gradient solar pond power cycle.

My role in this project:

The complete design project calls for an analysis of the individual components, the system, and the economics of the system. My role focused on writing computer code in MATLAB capable of analyzing the entire cycle. The codes consisted of subroutines capable of analyzing the thermodynamic properties of seawater and pure water, as well as the psychrometric properties of atmospheric air. Other subroutines combined these subroutines to analyze and size the separator, turbine, condenser, cooling tower, pump, and pipes. Codes were also written to evaluate the entire cycle and its components for a given flow rate of water out of the pond, and to estimate the efficiency of the power cycle. In addition, I contributed to the final writing and presentation of the design report.

Bibliography

1. Tabor, H. "Review Article: Solar Ponds." *Solar Energy*, 27, No. 3 (1981): 181- 194.

2. Dr. Robert L. Reid. Class Lecture. January 16, 2001.

3. Hull, John R. and Carl E. Nielson. Solar Pond Research and Development. *Solar Collectors, Energy Storage, and Materials*, Francis de Winter, Ed. Volume 5 of Solar Heat Technologies: Fundamentals and Applications, Charles A. Bankston, editor-in-chief. The MIT Press. Cambridge, MA: 1990.

4. Kut, David and Gerard Hare. *Applied Solar Energy*, 2nd Ed. Butterworth's, London: 1983.

5. Moran, Michael and Howard Shapiro. *Fundamentals of Engineering Thermodynamics*, 3rd Ed. John Wiley & Sons, Inc. New York: 1996.

Analysis of a Power Generation System Utilizing a Salt Gradient Solar Pond

Charlotte Walker

April 30, 2001

Abstract:

Salt gradient solar ponds are an environmentally friendly energy resource, capable of providing 60 - 85°C water for process heating, desalination, and energy generation. The possibility of generating power from a salt-gradient solar pond is explored, using a flash separator and the pond's brine as the working fluid for the power cycle. The current design focuses on a 5 MW peaking power plant to be stationed in Bakersfield, California.

The design suggests that the flash separation process will be capable of generating 0.0075 kg steam/kg brine, and will require a 14 m diameter and 42 m long flash drum operating at approximately 28700 Pa. The turbine will need to produce 7 MW of power in order to run the pumps, fans, vacuums associated with the process. The condenser and cooling tower will need to be able to handle 97.6 m^3/min (23,148 gal/min) of water, which is on par with the flow rate of the Tennessee River. The turbine and condenser will also need to be placed 12 m above the surface level of the pond, and the pumps will need to be buried 5 m below the separator. The overall first law efficiency of the power cycle is around 5%, compared with a Carnot efficiency of around 14%. The initial estimate of the pond area suggests that 2,500,000 m^2 (620 acre) should provide enough thermal energy for the pond to operate at peek efficiency for 8 months of the year.

The design analysis suggests that while the system is theoretically possible, it may be physically impractical and economically unsound.

Design of a Power Generation System Using a Salt-Gradient Solar Pond

Mechanical Engineering 479
Mechanical Engineering Senior Design

By:

Brett Lee (Team Leader)
John Camp
Mike Orr
Charlotte Walker

Submitted to:

Dr. Robert J. Krane

May 8, 2001

TEAM
Dougherty Engineering Building
Knoxville, TN 37996
(865) 974-6806

May 8, 2001

Dr. Robert J. Krane
209 Dougherty Engineering Building
Knoxville, TN 37996-2216

Dear Dr. Krane,

The enclosed report details the design of a power generation system utilizing a salt-gradient solar pond to be situated in Bakersfield, CA. The design report covers both the technical and economic aspects of the design.

Although some aspects of the design may require some additional attention, like designing a separator for sub-atmospheric conditions since we were unable to procure a commercial model, the design represents a fully functional power plant, with a 5.21 MW capacity.

The economic analysis shows that the plant is profitable under the design conditions. Selling electricity at $ 0.12 per kilowatt-hour allows the recovery of the initial investment cost of $120 million over a period of 20 years. The life of the power plant is assumed to be 30 years.

If you should have any questions about the design of the salt-gradient solar pond, please feel free to contact us. We have enjoyed this opportunity to work with you on this project, and hope that you will consider our services for any future work.

Sincerely,

Brett Lee – Team Leader

John Camp Mike Orr Charlotte Walker

Enclosures (1)

Table of Contents

Executive Summary

The challenge for the TEAM was to design a power generation system using a salt-gradient solar pond as the heat source. The power system was to be a 5 MW producing facility serving a plant located just outside of Bakersfield, California. The power plant was to be based on a flash cycle that used pond brine as the working fluid.

During the design of the steam power plant, the TEAM developed necessary the MATLAB code to determine the operating specifications of each piece of equipment. The equipment included a separator, a steam turbine, a condenser, a cooling tower, pumps, and pipes.

Amidst the current energy crisis and demand for renewable energy utilization in California, development of alternative technologies at the local and regional levels is abounding. The salt-gradient solar pond is one such technology that seems to be both relevant and promising for California. By selling electricity at the rate of $ 0.12 per kilowatt-hour, TEAM concludes that the design of a power generation cycle utilizing a salt-gradient solar pond is possible. Recovery from the initial investment of $120 million to construct the pond and the power plant will occur 20 years into the life of the plant, estimated to be 30 years. The proposed design does present a feasible alternative energy source.

The final recommendation from TEAM is to continue research efforts until a working system, complete with sized equipment, is readily available. This would require concentrated analysis of the separator in order to meet the design specifications.

Nomenclature

Symbol	Representing	Units
$\$_{now}$	Monetary value in the year of interest (current year)	dollars
$\$_{then}$	Monetary value at reference year	dollars
ε	Equivalent roughness of pipe, for steel this value is 0.045×10^{-3}	m
η	First law efficiency of the power cycle	
η_p	Pump efficiency	
η_t	Turbine efficiency	
λ	Latent heat of vaporization	J/kg
μ	Viscosity	$N \cdot sec/m^2$
μ_G	Viscosity of the vapor mixture leaving the condenser	$N \cdot sec/m^2$
υ	air velocity over pond	ft/min
ρ	Fluid density	kg/m^3
ρ_G	Fluid density of thevapor mixture leaving the condenser	kg/m^3
$\rho_{L(v)}$	Fluid density of a liquid (vapor)	kg/m^3
σ	Surface tension	N/m
Δu_L	Incremental drop velocity	m/s
Δp	Pressure change	Pa
ΔP_n	Pressure change across nozzle	Pa
Δt	Increment time	s
ΔT_L	Drop temperature rise	K
Δz	Increment length	m
a	Contact area per tower volume	m^2/m^3
A	Cross-sectional area of the pipe	m^2
A	Area of the pond surface, used in water evaporation correlation	ft^2
A_t	Annual income in excess of annual cost	dollars
Area	Area of the pond surface	m^2

B_t	Annual income at the end of year t	dollars
c_p	Specific heat	J/kg·K
c_{pG}	Specific heat of the vapor mixture leaving the condenser	J/kg·K
c_{pL}	Specific heat of the liquid in the condenser	J/kg·K
c_{pW}	Specific heat of water in Merkel Equation	J/kg·K
C	Cost of equipment	dollars
C_r	Reference cost of equipment	dollars
C_t	Annual cost at the end of year t	dollars
CPI_{now}	Consumer price index for the year of interest (current year)	
CPI_{then}	Consumer price index for reference year	
d	Mean drop size	M
D	Pipe diameter	M
D_{AB}	Diffusivity	M^2/s
D_v	Vessel diameter	M
E	Entrainment allowance in condenser	kg liquid/kg vapor
f	Flash fraction, mass of steam per mass of brine flashed	
f	Friction factor	
F	Flow number	
h_a	Actual head rise of fluid	m
h_{fg}	Latent heat of vaporization at water surface temperature	Btu/lb
h_G	Vapor heat transfer coefficient	W/mK
h_L	Losses in the fluid system due to friction and fittings	m
h_L	Major losses in pipe	m
h^F_{mix}	Enthalpy of the feed line, the feed is a mixture of salt and water	J/kg
h^L_{mix}	Enthalpy of the liquid leaving the flash drum, the liquid is a salt and water mixture	J/kg

h_{pump}	Enthalpy of the brine leaving the separator and returning to the pond	J/kg
$h_{separator}$	Enthalpy of the fluid leaving the pond and entering the separator	J/kg
h_t	Enthalpy of the inlet steam to the turbine	J/kg
h_{t2}	Enthalpy of the exit steam of the turbine	J/kg
h_{t2s}	Enthalpy of the exit steam of the turbine, assuming the turbine behaves as an isentropic process	J/kg
h_{tower}	Enthalpy of the fluid exiting the cooling tower and entering the pond	J/kg
h_w^v	Enthalpy of the vapor leaving the separator, which is assumed to be pure water	J/kg
H_A	Enthalpy of air-water vapor mixture at inlet wet-bulb temperature in Merkel Equation	J/kg
H_W	enthalpy of air-water vapor mixture at bulk water temperature in Merkel equation	J/kg
I	Solar insolation (incident radiation)	$J/m^2/day$
g	Acceleration due to gravity, in SI units 9.81	m/s^2
k	Thermal Conductivity	W/m·K
k_G	Mass transfer coefficient	$kgmol /m^2.sec.(N/m^2)$
K	Mass transfer coefficient in Merkel Equation	
K_L	Loss coefficient for throttle valve	
l	Pipe length	m
m	Mass flux	kg/m^2 sec
m	Cost exponent	
$\dot{m}$	Mass flow rate	kg/s
$\dot{m}_{pump}$	Mass flow rate of the brine leaving the separator and returning to the pond	kg/s

$\dot{m}_{separator}$	Mass flow rate leaving the pond and entering the separator. This mass is also referred to as the feed.	kg/s
$\dot{m}_{tower}$	Mass flow rate exiting the cooling tower and entering the pond	kg/s
L_v	Vessel Length	m
n	Economic life-time of the pond	years
NPSH	Net positive suction head	m
$NPSH_A$	Available net positive suction head	m
NPW	Net present worth	dollars
$p_{1(2)}$	Pressure at position 1 (2) in pump analysis	Pa
$p_{1(2)}$	Pressure at point 1 (2) of the pipe	Pa
p_a	Saturation vapor pressure at dew point of ambient air, used in water evaporation correlation	in. Hg
$p_{condenser}$	Pressure in the condenser	Pa
p_{flash}	Flash pressure	Pa
p_{pond}	Pressure at the pond surface	Pa
p_s	Pressure on the suction side of the pump	Pa
p_v	Vapor pressure of the fluid in the pump	Pa
p_w	Saturation vapor pressure at temperature of surface water, used in water evaporation correlation	in. Hg
P_a	Partial pressure of condensing vapor	N/m^2
P_{ai}	Partial pressure of the condensing vapor at the interface	N/m^2
P_f	Power gained by fluid	W
PW	Present worth	dollars
q	Total heat flux	W/m^2
q_s	Sensible heat flux	W/m^2
Q	Heat transfer into the pond	W
Q	Volumetric flowrate	m^3/s
Q_v	Volumetric flow rate of the of the vapor	m^3/s

R	Gas Constant, equal to 8.314	kg/kgmol.K
Re	Reynold's number, $Re = VD/\nu$, where ν is the kinematic viscosity of the flowing fluid (m^2/s)	
S	Size of equipment	varies
S^F	Salinity of the feed line from the pond	g/kg
S^L	Salinity of the liquid leaving the flash drum	g/kg
S_r	Reference size of equipment	varies
t	Specific year of interest in the life cycle of the pond	year
T_e	Outlet water temperature of cooling tower	K
T_{flash}	Flash temperature	K
T_i	Inlet water temperature of cooling tower	K
T_G	Bulk vapor temperature	K
T_I	Temperature at the liquid/vapor interface of the condenser	K
T_{pond}	Pond Temperature	K
u_{Lin}	Initial drop velocity	m/s
$\bar{u}_L$	Average drop velocity	m/s
u_t, u_s	Settling velocity	m/s
V	Fluid velocity	m/s
V	Active cooling volume per plan area in Merkel equation	m^3/m^2
$V_{1(2)}$	Velocity at position 1 (2) in pump analysis	m/s
$V_{1(2)}$	Velocity at point 1 (2) of the pipe	m/s
V_{avg}	Average velocity in the pipe	m/s
$V_{condenser}$	Velocity of the fluid leaving the condenser	m/s
V_{pond}	Velocity of the fluid at the pond surface	m/s
V_s	Fluid velocity on the suction side of the pump	m/s
$W_{condenser}$	Energy needed by the pump to raise the water from the	W

	cooling tower to the condenser	
W_{net}	Net work done by power cycle	W
w_p	evaporation rate of water	lb/hr
W_{pump}	Energy needed by the pump used to return the brine to the pond after it has been through the separator	W
$\dot{W}_t$	Power produced by the turbine	W
W_{tower}	Energy needed by the cooling tower to run its fans	W
$W_{turbine}$	Energy created by the turbine	W
W_{vacuum}	Energy needed to run the vacuum pump attached to the condenser and used to exhaust inerts	W
z	Vertical height of fluid, measured to some arbitrary reference plane	m
$z_{1(2)}$	Elevation at position 1 (2) in pump analysis	m
$z_{1(2)}$	Height at point 1 (2) of the pipe	m
$z_{condenser}$	Elevation of the condenser relative to an arbitrary reference plane	m
z_{pond}	Elevation of the pond relative to an arbitrary reference plane	m

I. Introduction

A. Salt Gradient Solar Pond

The sun is the most abundant source of renewable energy, making it one of the best alternatives to the non-renewable sources of energy. One way to harness this solar energy is through the use of solar ponds. Solar ponds are large-scale energy collectors with integral heat storage for supplying thermal energy. They can be used for various applications, such as process heating, water desalination, refrigeration, drying and power generation. For the purpose of this design, the focus of solar pond applications is obviously power generation.

The salt-gradient solar pond works on a very simple principle. In a normal reservoir, the differential heating would set up natural convection currents by which the heated water rises to the surface. The heat is lost to the ambient surroundings by evaporation, convection and radiation. The net result is that the pond water remains at, or near, the atmospheric temperature. The salt-gradient solar pond suppresses natural convection in the pond by dissolving salt in the bottom layer of the pond making it too dense to rise even though its temperature increases.

A salt-gradient solar pond is a large reservoir of saline water, with the exception that a specific salinity (or density) profile is artificially created and maintained in the pond. Typically, a solar pond consists of three zones: an upper convective zone (UCZ) with a uniform, low density; a non-convective zone (NCZ) with a gradually increasing density; and a lower convective zone (LCZ), also called the storage zone, with a uniform high density, as shown in Figure I-1.

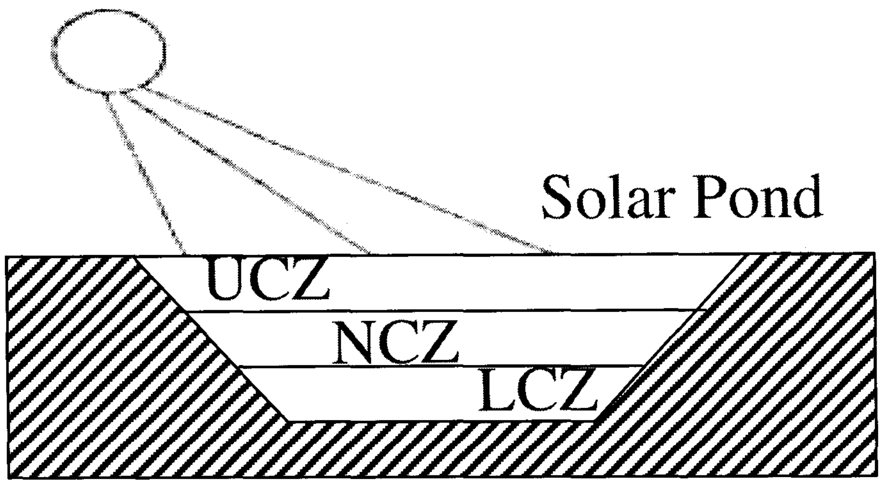


Figure I-1: **Schematic of a solar pond**

The surface zone, or UCZ, is at atmospheric temperature and has little salt content. The bottom zone is very hot, 70–85 ° C, and is very salty. It is this zone that collects and stores solar energy, and is, therefore, known as the storage zone. Separating these two zones is the important gradient zone or NCZ. Here the salt content increases as depth increases, thereby creating a salinity or density gradient. This gradient zone acts as a transparent insulator permitting sunlight to reach the bottom zone but also entrapping it there. The trapped (solar) energy is then withdrawn from the pond in the form of hot brine from the storage zone.

Although salt-gradient solar ponds can be constructed anywhere, constructing the ponds is only economical in locations where salt cost is low, a sufficient supply of sea water or water for filling is available, high solar radiation occurs, and land is available at low cost.

B. Why Use Salt Gradient Solar Ponds?

The fundamental purposes for utilizing salt-gradient solar ponds for power generation revolve around environmental issues arising from the use of conventional fuels. First, energy is produced without burning fuel, which reduces the pollution factor. Second, conventional energy resources are conserved.

Salt-gradient solar ponds are essentially low cost solar collectors with integrated storage, and hence are potential, cheaper alternatives to flat plate collector systems in suitable locations. A salt-gradient solar pond is a large reservoir of saline water, with the exception that a specific salinity (or density) profile is artificially created and maintained in the pond. The pond is also kept as transparent to solar radiation as possible by periodically treating it for algae and dust control (1).

The solar radiation penetrating the pond is absorbed in the different layers and causes the temperature to increase. Because water is a poor conductor of heat, loss by conduction from the lower zone to the upper zone is low. The large mass of saline water in the lower zone thus gets transformed into a large thermal storage, from which heat can be extracted for useful purposes. Once the pond is heated up (which takes 2–3 months after establishing the salinity gradient), the temperature change in the storage zone is controlled by (a) the solar radiation flux reaching the zone, (b) the conductive heat loss to the earth and through the non-convective zone, and (c) the amount of heat extracted for useful purposes.

Salt-gradient solar ponds have a low capital cost owing to the fact that they are based on low cost materials like clay, plastic and salt. However, solar ponds receive less amounts of solar radiation than other types of collectors because they cannot be angled to absorb the maximum amount. Thus, the operating efficiency of solar ponds is lower in comparison to conventional flat plate collectors. In fact, solar pond areas ranging from 2000-250,000 m^2 can provide around 0.20-5 MW. The operating efficiency of solar ponds is about 2-3% for electricity generation, and 15-30% when the desired ouput is thermal energy (2).

The advantages of low initial costs and no fuel costs offset the disadvantages of lower efficiency. Solar ponds cannot be installed on roof tops, require larger areas for the same heat delivery rates, require salt and water, and require trained persons for operation and maintenance. There is a critical size below which it is infeasible to operate the pond effectively. The requirements of land, salt and water suggest that solar ponds are better constructed on wastelands, on desert lands, or close to salt works.

C. Incorporating the Salt Gradient Solar Pond in Bakersfield, CA

The location for the salt-gradient solar pond is Bakersfield, CA. The facility of the customer is located just outside of Bakersfield. Bakersfield has a year-round availability of solar energy, the dry and hot climate, and the availability of water resources. The average high/low temperatures for Bakersfield are 78/51°F. The number of sunny days per year averages 273 days. See Table I-1 for a complete listing of the high and low temperatures as well as other environmental data for Bakersfield. As far as water resources are concerned, the California State Water Project (SWP) contains an extensive network of reservoirs, aqueducts, power plants and pumping stations. The main function of the SWP is to manage water supply, storing surplus water during wet periods and distributing it to service areas throughout California. The Federal Central Valley Water Project contributes to the water needs of power plants, too. However, existing SWP facilities can supply approximately 2.4 million acre-feet of water each year. This system could ultimately be expanded to provide 4.2 million acre-feet per year. This difference of 1.8 million acre-feet alone convinces the TEAM that enough water could be supplied to install a salt-gradient solar pond (3).

Table I-1. **Environmental Data for Bakersfield, California**

Latitude: 35° 25″ N

Longitude: 119° 03″ W

Elevation: 495 ft

Month	Average High Temperature, F	Average Low Temperature, F	Extreme High Temperature, F	Relative Humidity (7 am), %	Windspeed (knots) and direction	Solar Radiation, $MJ/m^2/day$
January	57	38	82	86	5 E	8.70
February	64	42	87	82	5 E	12.52
March	69	45	92	75	8 NNW	18.11
April	76	50	101	63	9 NW	23.79
May	84	57	107	52	9 NW	28.49
June	92	64	114	46	9 NW	31.22
July	99	70	115	44	9 NW	30.48
August	96	68	112	50	8 NW	27.50
September	91	63	112	56	8 NW	22.62
October	81	54	103	63	7 NW	16.56
November	67	45	91	77	5 E	10.70
December	57	38	83	86	5 E	7.69

[Weather data from http://www.bestplaces.net (4) . Solar radiation data from PONDFEAS input (5).]

D. Power Generation System

The electrical power generation system is a 5 MW peeking facility servicing a plant located just outside of Bakersfield, CA. The system was based on a flash cycle that uses the pond brine as the working fluid. The flash distillation process was chosen because it was a known, simple, and successfully proven cycle.

E. Flash Cycle

The following segment was excerpted from "Open Cycles for Power Generation from Solar Ponds," a paper presented by Kornhauser at Forum 2001 in Washington, D.C. in April (6).

There are three types of open cycles that appear suitable for use with solar ponds:

1. a flash cycle, in which low pressure steam is produced by throttling of hot pond brine.
2. a gravity expansion cycle, in which low pressure steam is produced by isentropic expansion of brine in a bubble lift pump.
3. an ejector expansion cycle, in which low pressure steam is produced by isentropic expansion of brine in the motive nozzle of an ejector.

Although the flash cycle was expected to have the lowest efficiency due to the inherent losses in the flashing process, it was chosen for the power plant design because it represented the simplest cycle of the three open cycle options. It also uses no untried technology.

In a flash cycle, low pressure steam is produced by throttling hot pond brine.

1. Hot pond brine is pumped from the bottom of the pond and throttled to low pressure. The brine is flashed into steam and a more concentrated brine.
2. Then the brine enters a flash separator, in which the steam is separated from the concentrated brine. The separator pressure is the saturation pressure corresponding to the desired brine return temperature. A demister is used to reduce droplet carryover.
3. The liquid brine is then pumped back to the pond bottom. Locating the separator as high as possible, without causing flashing upstream of the throttle, will minimize the required pump head. Pump head is the difference between saturation pressure at brine supply conditions and saturation pressure at brine return conditions.
4. The steam is expanded to condensing pressure through a turbine.

5. Expanded steam is condensed and returned to the pond surface. By locating the condenser high enough above the surface, this return flow can be obtained without pumping. Returning pure water to the pond surface helps maintain the pond's salinity gradient. The flash cycle is represented in Figure I-2.

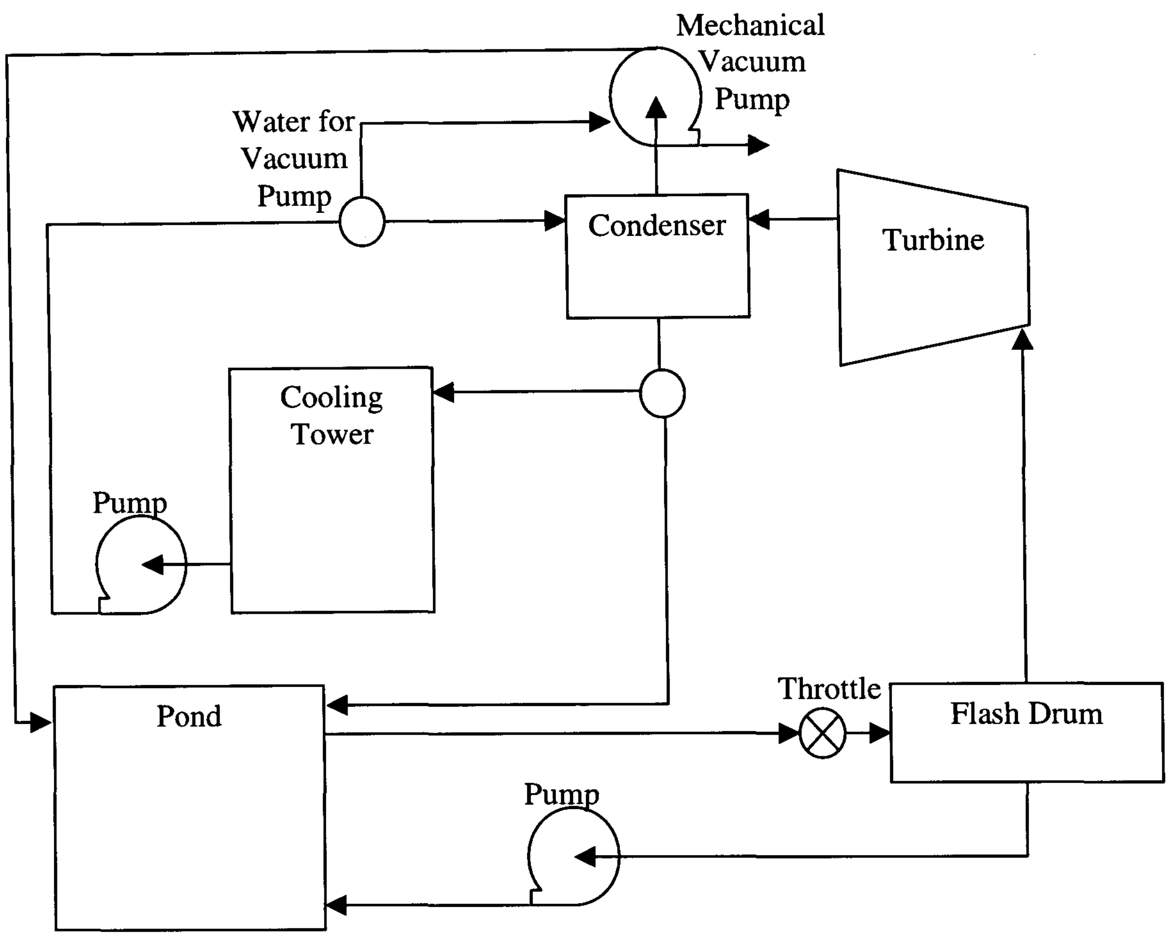


Figure I-2. **Flash cycle for salt-gradient solar pond power generation.**

After thoroughly understanding the entire flash cycle, the TEAM was ready to begin the preliminary design phase and started analyzing the components of the flash cycle. The equipment needed to produce a steam power plant is detailed later in this report but includes pumps, a separator, a turbine, a condenser, a cooling tower, and a mechanical vacuum pump.

II. Design Analysis

A. Introduction

In order to evaluate the feasibility of the selected design scheme, a detailed analysis of the power system must be performed. This analysis includes designing unforeseen system components and processes, in addition to selecting process methods and specifying operating conditions. The following discussions outline the analysis techniques, assumptions, and selection criteria used to analyze the system.

B. Flash Separator

One of the key components of the salt-gradient solar pond's power cycle is the flash separator, where the low-pressure steam for the turbine is produced. The separator is made up of two components, a throttle valve and a flash drum, which work together to generate and separate the steam from the pond's hot brine.

First, the hot brine flows throw the throttling valve, which is an isenthalphic and adiabatic process. The pressure drop across the throttle results in the brine entering the two-phase region where both steam and brine coexist in the working fluid, as shown in Figure II-1.

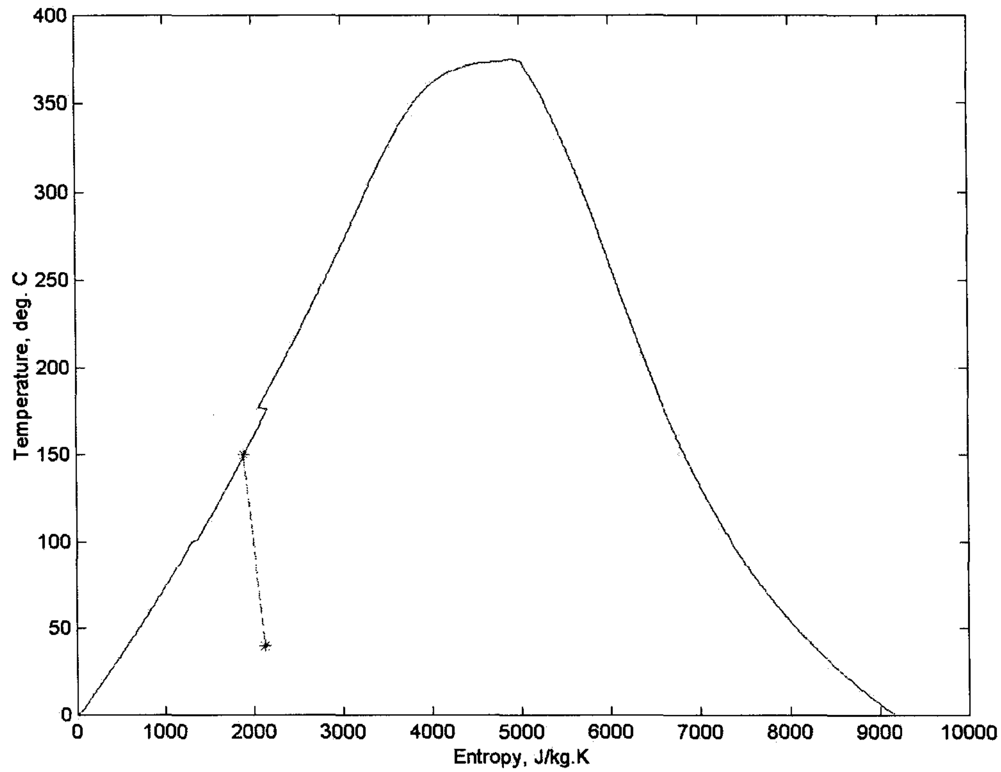


***Figure II-1*: Illustration of an isenthalpic process on a Temperature-Entropy Diagram.**

Then the two-phase mixture is sent into the flash drum, where the different phases are separated. Upon entering the flash drum, gravity pulls the more dense liquid down to the bottom of the drum and the vapor rises to the top of the drum. The vapor is then passed through a demister pad to remove the liquid droplets from the vapor and to the turbine.

The design process for the flash separator consists of 3 steps. First the throttle characteristics must be specified, next the inlet and exit conditions of the separator must be determined, and finally the separator must be sized.

The throttle characteristics can be specified from the pipe geometry and the pressure change required across the throttle. If Bernoulli's Equation, modified for losses, is used to model the flow in the pipe we have:

$$\frac{p_1}{g\rho}+\frac{V_1^2}{2g}+z_1=\frac{p_2}{g\rho}+\frac{V_2^2}{2g}+z_2+K_L\frac{V^2}{2g}. \qquad \text{(II-1)}$$

However, it is reasonable to assume that the pipe diameter is the same on either side of the valve and that there is not an appreciable change in height, so that

$$\begin{aligned} V_1 &= V_2 = V \\ z_1 &= z_2 \end{aligned}. \qquad \text{(II-2)}$$

Rearranging the equation, the throttle's loss coefficient equals:

$$K_L=\frac{2(p_1-p_2)}{V^2\rho}=\frac{2(p_1-p_2)A^2\rho}{\dot{m}^2}, \qquad \text{(II-3)}$$

where $\dot{m}=\rho A V$ is the mass flow rate. Knowing the throttling coefficient is not enough to design a throttling valve, however, it will guide the selection of which types of valves will be most appropriate to the application.

For the analysis of the flash cycle, an "Equilibrium Assumption" was made for the exit conditions of the flash drum, which said that the temperature, pressure, and chemical potential of the exiting vapor would be equal to that of the exiting liquid (7). However, the presence of salt in the brine causes the boiling point of the solution to rise above that of pure water, which is known as the boiling point elevation. For a given flash pressure, the flash temperature can be determined to be the saturation temperature of pure water at the flash pressure plus the boiling point elevation due to the salinity of the feed.

Once the temperature and pressure at the exit is determined, the masses are balanced and the First Law of Thermodynamics is applied. This leads to the iterative calculation shown in Table II-1 to determine the value of the flash fraction, *f*; where the flash fraction is defined as the mass flow rate of the water vapor divided by the mass flow rate of the feed.

***Table II-1*: Iterative calculation used to determine the flash fraction.**

1. Estimate the flash fraction, *f*.	$f = \dfrac{\dot{m}_w^v}{\dot{m}_{mix}^F}$
2. Calculate the salinity of the exiting liquid, S^L, as a function of the salinity of the feed, S^F, and the flash fraction, *f*.	$S^L = \dfrac{S^F}{1-f}$
3. Calculate the enthalpy of the exiting liquid, h_{mix}^L, from the correlations numbered 22.25/26 in Kahn (8).	$h_{mix}^L = h_{mix}^L\left(T_{flash}, S^L\right)$
4. Calculate the enthalpy of the feed, h_{mix}^F, from correlations numbered 22.25/26 in Kahn (8).	$h_{mix}^F = h_{mix}^F\left(T_{pond}, S^F\right)$
5. Calculate the enthalpy of the exiting water vapor, h_w^v, from equation 2 of Badr (9).	$h_w^v = h_w^v\left(T_{flash}, p_{flash}\right)$
6. Recalculate the flash fraction from the 1st Law Analysis.	$f_{new} = \dfrac{h_{mix}^F - h_{mix}^L}{h_w^v - h_{mix}^L}$
7. If the new flash fraction is sufficiently close to the old flash fraction, then quit; else let $f = f_{new}$ and continue from step 2.	

Once the fluid conditions are known, it is necessary to size the separator's flash drum. The sizing analysis follows that presented in Sinnott (10), and can be applied to two types of flash drums: vertical and horizontal. Both geometries use the settling velocity of the liquid droplets to size the vessel, which is given by:

$$u_t = 0.07\sqrt{\frac{\rho_L - \rho_v}{\rho_v}}, \tag{II-4}$$

where the settling velocity, u_t, is given in m/s and the densities are given in kg/m^3. If a demister pad is used, the above relationship is used to calculate the height of the vessel; however, if no demister pad is used, u_t should be multiplied by 0.15 to give an appropriate margin of safety.

The diameter of a vertical separator is sized separately from the height, and is given by:

$$D_v = \sqrt{\frac{4V_v}{u_s \pi}}, \tag{II-5}$$

where V_v is the volumetric flow rate of the of the vapor and u_s is equal to u_t when a demister pad is employed. The height of the vessel is the summation of 4 parts. First, the depth of the liquid, if a 10 minute hold-up time is allowed, second the height from the liquid to the feed inlet, third the distance from the feed to the demister pad, and finally the distance from the demister pad to the vapor exit. These distances are shown in Figure II-2.

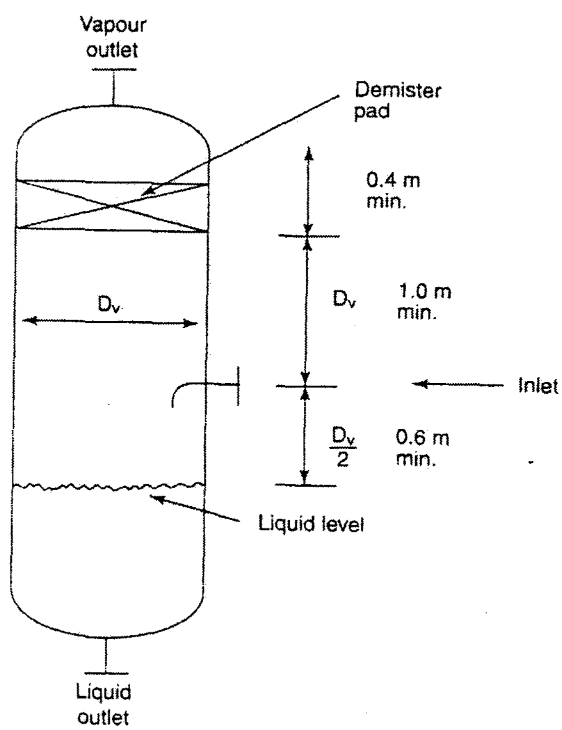


***Figure II-2*: A typical vertical separator configuration [reproduced from Sinnot (10)].**

A horizontal separator is slightly more difficult to size since the length and diameter cannot be sized separately. The optimum length to diameter ratio is dependent upon the operating pressure of the separator, as shown in Table II-2.

Table II-2: **General guidelines for length to diameter ratios for horizontal separators, [reproduced from Sinnot p. 461(10)].**

Operating pressure, bar	Length: diameter, L_v/D_v
0-20	3
20-35	4
>35	5

Sinnot (10) recommends that the initial design assume that the liquid height will be equal to half of the diameter. First, the settling velocity of the liquid and the volumetric flow rate of the steam are determined. Next, since the time required for the gas to settle to the liquid surface is given by

$$\text{vapour residence time} = \frac{\text{liquid height}}{\text{settling velocity}} \qquad \text{(II-6)}$$

and the actual residence time is given by

$$\text{actual residence time} = \frac{\text{vessel length}}{\text{vapour velocity}} \qquad \text{(II-7)}$$

it is possible to calculate the required diameter based on the vapor velocity. This diameter is then compared to the diameter required for a 10-minute hold-up time for the liquid, and the greater diameter is used for the vessel. An illustration of a typical horizontal separator may be found in Figure II-3.

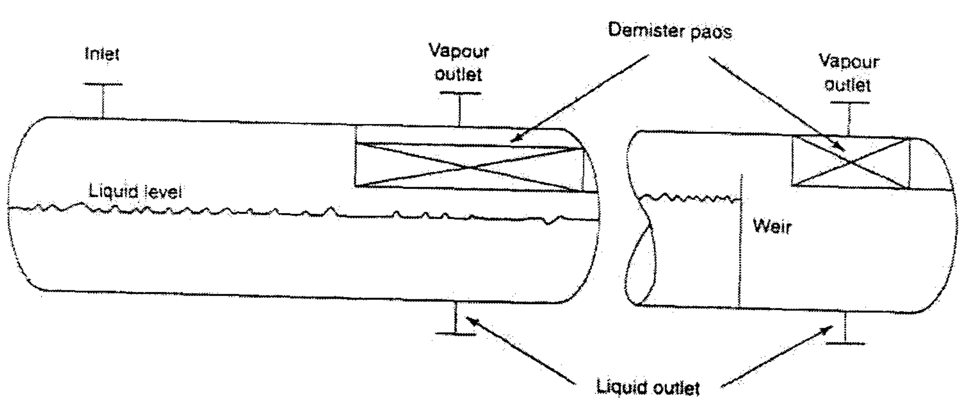


Figure II-3: **A typical horizontal separator configuration [reproduced from Sinnot (10)].**

C. Pumping Systems

The main purpose of pumps is to increase the pressure of a fluid. Pumps can be divided into two categories: positive displacement machines and turbomachines. Both types are widely used, from window fans (a turbomachine) to bicycle pumps (a positive displacement machine). The following sizing analysis will deal primarily with centrifugal pumps, a type of turbomachine.

Pumps are sized based the required pressure rise of the fluid and the volumetric flowrate. The pressure rise of the fluid is often expressed in terms of a length measurement, or "head," which is related to the power actually transferred to the fluid by the equation

$$P_f = \rho g Q h_a, \tag{II-8}$$

where h_a is the actual head rise of the fluid. h_a is given by the system curve

$$h_a = \frac{p_2 - p_1}{\rho g} + z_2 - z_1 + \frac{V_2^2 - V_1^2}{2g} + \sum h_L. \tag{II-9}$$

Since the pressure and elevation differences are usually set by system requirements, the system curve represents the amount of power necessary to generate the desired change of state for a given system, and is generally a function of the flowrate squared.

Once the head rise and flowrate for a given application are known, a particular pump can be selected from manufacturer's performance curves. The manufacturer will then specify the pump's efficiency and brake horsepower requirements. The pump's efficiency is given by

$$\eta_p = \frac{\text{power gained by fluid}}{\text{brake horsepower}}, \tag{II-10}$$

where the brake horsepower is the total shaft power driving the pump. Typical pump efficiencies are between 60% and 80%, depending on the application. For this project, pump efficiencies were assumed to be 80%.

Pumps may also be sized to be used in series or parallel if a single pump cannot provide both the required head and flowrate. Pumps connected in series add their head capabilities while pumps used in parallel add their flowrates, as shown in Figure II-4. The actual performance of the pump may not actually be increased linearly with the addition of more pumps since the performance of a pump is based on the intersection of the system curve with the performance curve.

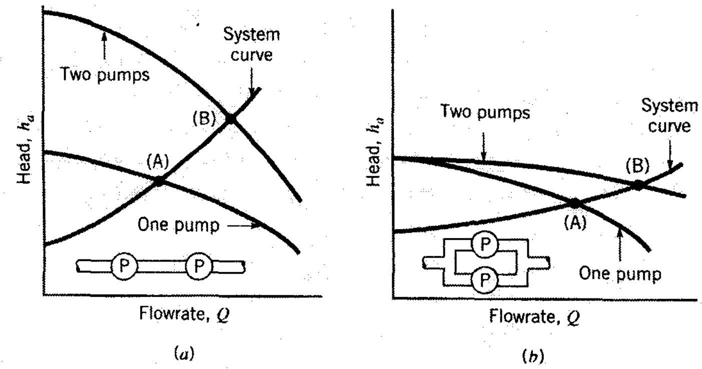


Figure II-4**: (a) The effect on the pump curve for operating pumps in series is to add the head rise at a given flowrate. (b) The effect on the pump curve for operating pumps in parallel is to add the flowrates for a given head rise. [Reproduced from Munson (11)].**

In the design of a fluid system, one concern is cavitation, which is the vaporization of part of the system's fluid due to the system dropping below the vapor pressure of the fluid. To determine if it will occur, it is necessary to calculate the net positive suction head available and compare it to the net positive suction head that is required. The available net positive suction head is given by

$$\mathrm{NPSH}_{\mathrm{A}} = \frac{p_s}{\rho g} + \frac{V_s^2}{2g} - \frac{p_v}{\rho g}. \tag{II-11}$$

However, the required net positive suction head is usually determined by experimentation and should be provided by the manufacturer. The design should insure that the available suction head is greater than the required suction head to avoid cavitation, and the structural damage it can cause.

Mechanical vacuum pumps are sized similarly to ordinary fluid pumps; however, their sizing criterion consists the required volumetric flow rate of the gases, volume of the system, pump down time required, gas load, cost concerns, and the degree of vacuum needed (http://www.usvacuumpumps.com/pump_sizing.html, 12).

D. Turbine/ Generator

A turbine converts the enthalpy of a working fluid (in this case the low-pressure steam flashed from the hot pond water) into mechanical energy, in the form of a rotating shaft. High-pressure steam is expanded through nozzles at the turbine inlet. These nozzles convert the steam's high pressure into increased velocity. The shaft is rotated by this high velocity steam impacting small vanes attached along the outside of large wheels fitted to the shaft. As electrical energy is the desired output of the plant, the rotating shaft drives an electric generator.

The turbine process is limited by conservation of energy. However, with the reasonable assumption that changes in velocity, height, etc. are small compared to the enthalpy change of the steam, the process can be modeled as:

$$\dot{W}_t = \dot{m}(h_{t1} - h_{t2s}) \qquad \text{(II-12)}$$

For our design procedure, the exit state of the steam was determined by first fixing the isentropic exit state for a given steam quality. (It is not uncommon for commercial steam power cycles to have qualities as low as 90% at the exit of the turbine (13). For this project, the quality was assumed to be 94%, since this value gives a reasonable pressure drop across the turbine.) The isentropic exit properties were then determined by locating the state at which the steam has the same entropy as the inlet steam.

However, this process obviously cannot exist in the isentropic form it is modeled above. Energy conversion losses can occur in the form of friction, such as the steam moving in the nozzles and across the blade channels, or as the discs rotating through steam particles. In addition, losses can occur due to partial admission of steam between blades not being enacted by a nozzle group, or residual steam velocity that has not been fully utilized, etc. Likewise, losses related to conversion of this mechanical energy into electrical energy by the generator must be accounted for.

As a result, the turbine must be modeled as operating at some manufacturer specified efficiency η_t. Based on this overall efficiency, the actual enthalpy of the steam at the turbine is determined, and the power output is evaluated as:

$$\dot{W}_t = \eta_t \dot{m}(h_{t1} - h_{t2s}). \tag{II-13a}$$

This is because the turbine efficiency is defined as

$$\eta_t \equiv \frac{h_{t1} - h_{t2}}{h_{t1} - h_{t2s}}. \tag{II-13b}$$

This project assumed a turbine efficiency of 80%.

The desired net power output of the plant is 5MW. Therefore the turbine should be able to produce this output, plus the power needed to operate various processes such as the brine feedback pumps, the vacuum pump, etc. The power produced by the turbine is affected by the steam flow rate, and by the enthalpy change.

The enthalpy change is limited by the flash pressure of the steam leaving the separator. Therefore, increased flash pressure will allow more power generation for a given mass flow rate of steam. However, increasing the flash pressure decreases the flash fraction, and thus the mass flow rate of the steam, which decreases power generation. Increasing the feed flow rate to allow a constant steam flow rate will increase the parasitic pumping power needed to return the unflashed brine to the pond. As a result, selection of the most appropriate turbine operation point becomes an exercise in balancing flow rate with flash pressure. This balance is further addressed in the final design discussion.

E. Condenser

1. Condenser Design Background

The flash process used in the power cycle for the generation of electricity requires the use of a steam condenser at the exit of the turbine. The condenser is a device that converts steam into water using a coolant fluid. A condenser is a heat exchanger that removes heat from the steam by contacting it with a cooling fluid or cooling surface and reducing the steam to the saturation pressure at which point the steam changes to the liquid phase on the cooling surface or cooling droplets. The condensed water collects in the bottom of the vessel where it can then be recycled through the process once again [Graham (14), p. 551].

The primary reason for the use of a condenser is to maximize the net work and efficiency of the turbine. This occurs because the turbine power output is maximized when the outlet is discharged into a lower-pressure region. The lowest possible condenser pressure is the saturation pressure corresponding to the ambient temperature in the condenser. Also, a condenser is used for the purpose of recovering the working fluid so that it can be maintained in a closed loop system [Moran (15), p. 332].

There are two main types of condensers, the direct contact condenser and the surface condenser. The surface condenser uses an array of tubes ranging typically from about ½ to 1 inch. The coolant fluid flows though the pipes, which are within the steam region. The steam is cooled by contact with the surface of the pipes and steam condenses on this surface. One advantage of the surface condenser is that the steam is maintained at the same purity throughout the system, thus maintaining steady state throughout the system. A surface condenser's cooling fluid does not contact the steam, eliminating the chance of impurities in the system's steam source. The problems that arise with usage of the surface condenser are that excess power use, efficiency, and high maintenance costs are sacrificed for purity of the steam. Power is needed to pump the cooling fluid through numerous cooling tubes. Air leakage into the condenser is also a major problem. The use of cooling tubes can also lower the efficiency of the condenser because of the transfer of heat through an extra medium (the pipe material). The use fluid flowing through an array of pipes results in fouling of the inner walls, which can lead to high maintenance costs associated with the cleaning of each tube in the condenser. Plus, the fouling of cooling tubes also lowers the efficiency of the condenser [Marks (16), p. 9-62].

For the purpose of the salt gradient solar pond power cycle, the purity of the steam is not an issue because the condensate coming from the condenser outlet, which is pure water, will be placed back into the pond in order to maintain the temperature and salt concentration gradient. Thus, a direct contact spray condenser, which is more efficient and more economical than the surface condenser, will be used to condense the exiting steam from the turbine. This type of condenser uses a fine mist of water droplets to cool the steam to saturation pressure. This causes the steam to condense onto the water droplets, which then collect at the bottom of the tank. From here, the condensed water can be removed from the condenser vessel and pumped to the cooling tower [Azbel (17), p. 318].

A key requirement for maximum power output from the turbine is that the condenser must operate at sub-atmospheric pressure. Approximately one cubic foot of steam condenses into one cubic inch of water, thus creating a vacuum [Graham (14), p. 556]. The maintenance of a vacuum solely using the condensation process is impossible due to the inherent flaws in man-made machinery. Because the entire system is operating at sub-atmospheric pressure, there is a tendency for air to leak into the system where these flaws exist. This air can build up in the condenser, which can lead to the loss of vacuum conditions and the lowering of efficiency of the turbine. Also, air molecules blanket the steam-droplet interfaces. Another source of air leakage into the system is in the cooling water. Air is present in water and is drawn out of the water as it enters the vacuum [Marks (16), p. 9-65]. There are two common methods of removing the air build-up in the condenser. The first of these methods utilizes air ejectors. The second common method is the mechanical vacuum pump.

Ejectors are devices that remove non-condensable gases from the vacuum in the condenser. The ejector consists of a steam nozzle, a suction chamber, and a diffuser. The steam nozzle releases a high-velocity jet of steam into the diffuser creating suction in the suction chamber. The air inside the condenser is then entrained by the steam and carried out by the diffuser. Ejectors can be operated in series or as a single unit. They are low maintenance and have no moving parts, which accounts for their long life. The main drawback of the ejector is that it requires a substantial amount of steam to operate. For this reason, a mechanical vacuum pump may be suitable for the purpose of operating a solar salt gradient pond power system [Marks (16), p. 9-66].

The vacuum pump is used to eliminate non-condensable inerts from the condenser. The pump is not as reliable as the ejector but it is advantageous for the power cycle because it does not consume any steam from the steam source. This is the main reason that a vacuum pump will be used for the condenser.

2. Design Procedure Requirements

A design procedure for the direct contact condenser is given by Azbel (17, p. 342). The information needed to begin the design process is:

1. Temperature of the inlet and outlet liquid and vapor
2. Flow rate and compositions of inlet and outlet streams
3. Operating pressure, allowable pressure drop across spray section
4. Physical properties of the liquid and vapor (average values over the temperature range may be used) (listed below)
5. Saturation pressure and temperature for the condensing component
6. Spray nozzle characteristics (flow rate, nozzle pressure drop, mean drop size)
7. Entrainment allowance (kg liquid entrained/kg vapor vented)

The physical properties needed are:

specific heat (J/kg*°C), c_p

viscosity (N*sec/m^2), μ

thermal conductivity (W/m*°C), k

density (kg/m^3), ρ

latent heat (J/kg), λ

surface tension (N/m), σ

diffusivity (m^2/sec), D_{AB}

3. Design Procedure

Spray Nozzles: Select type and number of spray nozzles to produce required flow rate with the available nozzle pressure drop.

Mean Drop Size: The equation used for determination of the mean drop size (d) is:

$$d = 2.55\left(\frac{F}{\Delta P_n}\right)^{1/3} \qquad \text{(II-14)}$$

where F is the flow number and ΔP_n is the nozzle pressure drop.

Initial Drop Velocity: The equation for the initial drop velocity (u_{Lin}) is:

$$u_{Lin} = 0.8\left(\frac{2\Delta P_n}{\rho_l}\right)^{1/2} \qquad \text{(II-15)}$$

Length of Spray Section:

First, assume a change in drop velocity (Δu_L).

Calculate average drop velocity ($\overline{u}_L$) across the spray section length increment using the equation:

$$\overline{u}_L = u_{Lin} + \frac{\Delta u_L}{2} \qquad \text{(II-16)}$$

Calculate increment length (Δz) using the equation:

$$\Delta z = \frac{\overline{u}_L * \Delta u_L}{g - \left[(20.25) * \frac{B * \overline{u}_L^{1.16}}{d^{1.84}}\right]} \qquad \text{(II-17)}$$

where:

$$B = \frac{\rho_G}{\rho_L} * \left(\frac{\mu_G}{\rho_G}\right)^{0.84} \qquad \text{(II-18)}$$

Calculate the increment time (Δt) using the equation:

$$\Delta t = \frac{\Delta z}{\overline{u}_L} \qquad \text{(II-19)}$$

Calculate the vapor heat and mass transfer coefficients (h_G, K_G):

$$h_G = \frac{k_G}{d}\left[2 + 0.55\left(\frac{\overline{u}_L * \rho_G * d}{\mu_G}\right)^{\frac{1}{2}}\left(\frac{c_{pG} * \mu_G}{k_G}\right)^{\frac{1}{3}}\right] \qquad \text{(II-20)}$$

where:

$$k_G = \frac{D_{AB} * p}{d * R *}\left[2 + .0.55\left(\frac{\bar{u}_L * \rho_G * d}{\mu_G}\right)^{\frac{1}{2}} * \left(\frac{\mu_G}{\rho_G * D_{AB}}\right)^{\frac{1}{3}}\right] \quad \text{(II-21)}$$

Calculate total heat flux (q) using the equation:

$$q = q_S + m\lambda \quad \text{(II-22)}$$

where $m\lambda$ is the latent heat and q_S is the sensible heat flux found using:

$$q_S = \varepsilon\varepsilon h_G(T_G - T_L) \quad \text{(II-23)}$$

where

$$\varepsilon = \frac{C_0}{1 - \exp(C_0)} \quad \text{(II-24)}$$

$$C_0 = \frac{mC_{pG}}{h_G} \quad \text{(II-25)}$$

and

$$m\lambda = K_G(MW)(P_a - P_{ai}) \quad \text{(II-26)}$$

Calculate the drop temperature rise (ΔT_L) using the equation

$$\Delta T_L = \frac{\sigma * q * \Delta t}{c_{pL} * d * \rho_L} \quad \text{(II-27)}$$

Calculate the adjusted drop temperature ($\Delta T_{L_{n+l}}$) using the equation:

$$\Delta T_{L_{n+l}} = T_{l_n} + \Delta T_L \quad \text{(II-27)}$$

Continue iterating on the length of the condenser until the drop temperature exceeds the required drop temperature.

In order to determine the condenser inlet and outlet diameters and condenser vessel diameter, an entrainment allowance (E) is usually specified, but in this case, it must be assumed to be between 0.0001 and .05 kg liquid/kg vapor. For this design, an entrainment allowance of 0.05 kg liquid/kg vapor is arbitrarily assumed.

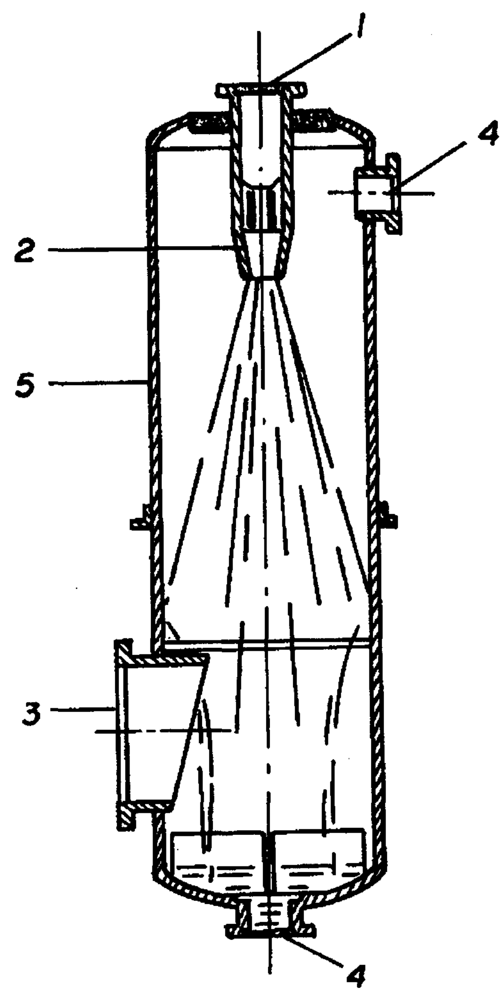


***Figure II-5*: Typical Counter-current Condenser; 1) Liquid Inlet 2) Spray Nozzles 3) Cover 4) Vent (Vapor Outlet) 5) Shell 6) Vent Cover [Reproduced from Azbel (17, p.319)]**

There are some important assumptions to be made while designing a condenser. First, the spray consists of rigid, spherical drops that are uniform in diameter that are traveling in the same downward direction of the length of the vessel. Secondly, the initial drop velocity will be the same for all drops. Also, the heat and momentum transfer at the nozzle inlet is ignored. It is assumed that the drops do not interact between one another. Finally, it is assumed that the vapor is well mixed throughout the vessel.

F. Height of Condenser and Turbine

One of the unique features of the flash cycle is its location of the condenser and turbine at a non-negligible height above the pond in order to eliminate the need for a pump, as shown in Figure II-5. In order to evaluate the height of the condenser above the pond surface, we will use Bernoulli's equation, modified for pipe losses to perform an energy balance on the system, as follows:

$$\frac{p_{condenser}}{\rho g}+\frac{V_{condenser}^2}{2g}+z_{condenser}=\frac{p_{pond}}{\rho g}+\frac{V_{pond}^2}{2g}+z_{pond}+\frac{1}{2g}\frac{f}{D}V_{avg}^2 \qquad \text{(II-29)}$$

However, because the flow is incompressible and the pipe diameter is constant, the velocity will be the same at the pond and the condenser. The equation then reduces to:

$$z_{condenser}-z_{pond}=\frac{p_{pond}-p_{condenser}}{\rho g}+\frac{1}{2g}\frac{f}{D}V_{avg}^2 \qquad \text{(II-30)}$$

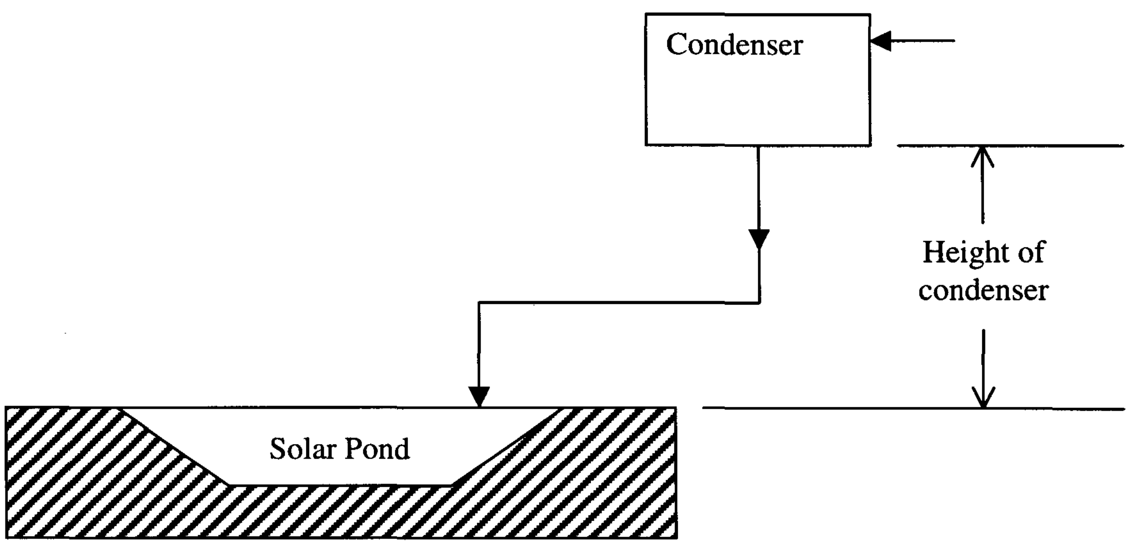


***Figure II-6*: Diagram showing vertical positioning of the turbine.**

G. Cooling Tower

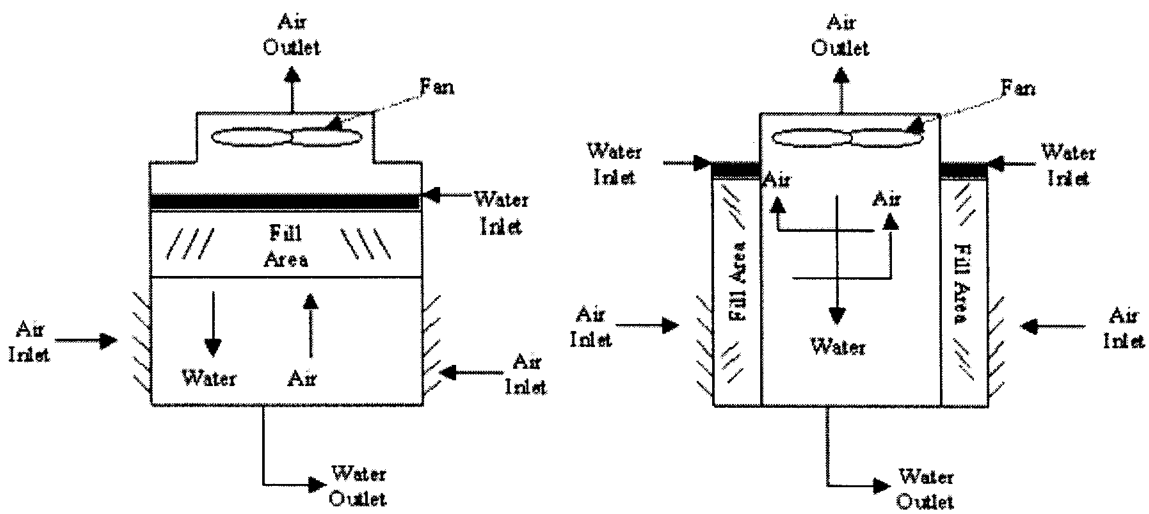
In order to recycle the condensate back into the condenser as cooling water, it will need to be cooled down by some means. However, not all of the condenser outlet water will be need to be cooled. Upon exiting the condenser, the warm water will be divided into two pipe leads, one going to the pond as warm water and the other to a cooling tower. The warm water would simply be placed back onto the pond surface, where it will mix with the proper level. In order to reuse the condensate in the condensation process, it must be cooled to the cooling water temperature in a cooling tower. There are two main types of cooling towers used in industry, a mechanical-draft cooling tower and a natural-draft cooling tower.

The natural-draft cooling tower is usually a very large and handles flow rates above 200,000 gallons/minute. This design utilizes concrete towers up to 500 feet high and 400 ft in diameter at the base. Air flows in at the bottom of the tower and is moved through the column by pressure difference between top and bottom of the tower (18).

Mechanical draft towers offer control of cooling rates by having adjustable fan diameters and speeds of operation. These towers often contain several modules (each with their own fan) called cells. The fan pulls air into the cooling cell, forcing it to come into contact with the warm water. The warm water drops downward over fill surfaces, or packing, which help increase the contact area and time of the water and the air. This helps maximize heat transfer between the two (18).

There are two versions of the mechanical-draft cooling tower. The first type is the forced-draft cooling tower (Figure II-7a). This design incorporates a fan conveniently located at ground level or at the top of the unit. Some problems that pertain to the forced-draft are that the air distribution is non-uniform near the inlet, there is some vapor recirculation from the discharge to the inlet, and limitations on fan diameter tend to occur.

The second type of fan-driven cooling tower is the induced-draft cooling tower (Figure II-7b). This type of cooling tower is predominantly used in the U.S. The fan is located on top of the cooling cell and at one of two places. In the counter-flow design, air enters the cooling cell at the bottom of the unit. In the cross-flow design, air enters the cooling cell uniformly from the sides of the unit. The cross-flow cooling tower design allows for more uniform air distribution than the counter-flow. Spray nozzles, splash plates, and downspouts allow for sufficient evaporation surfaces.



(a) (b)

***Figure II-7*: Mechanical Draft Cooling Towers. a) Forced-draft Cooling Tower b) Induced-draft Cooling Tower (Reproduced from 18.)**

The performance of a cooling tower is characterized by the cooling range of the tower and also by the approach of the tower. The cooling range of the tower is the total reduction of temperature of the water as it flows through the cooling cell. The approach of the tower is defined as the difference between the wet-bulb temperature of the air entering the tower and the temperature of the outlet water. These are graphically illustrated in Figure II-8.

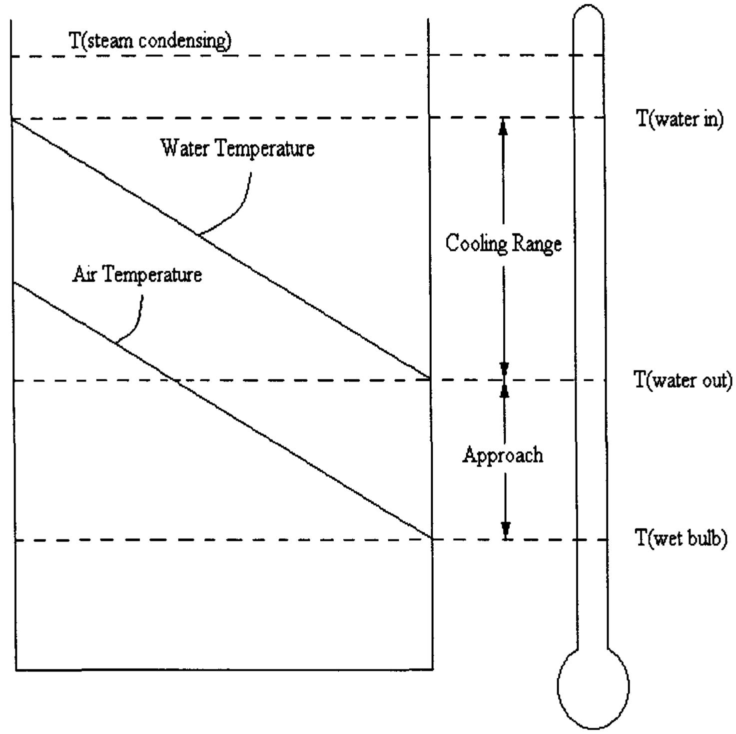


***Figure II-8*: Illustration of Cooling Range and Approach (Reproduced from Krane.)**

The heat transfer analysis of a cooling tower involves looking at a differential volume of the cooling cell. Assumptions made for this analysis include steady-state flow, and evaporation rate of water is negligible for the mass transfer balance. After analyzing heat and mass transfer in the differential volume, the resulting equation is:
where h_a, the actual head rise of the fluid, is given by the system curve

$$\frac{KaV}{L} = \int_{T_e}^{T_i} \frac{c_{pW} dT_W}{H_W - H_A}. \qquad \text{(II-29)}$$

This is a famous relationship known as the Merkel Equation. In the above equation, (K) is the mass transfer coefficient, (a) is the contact area per tower volume, (V) is the active cooling volume per plan area, ($T_{i,e}$) is the inlet and outlet water temperature, c_{pW} is the constant pressure specific heat of water, and ($H_{W,A}$) is the enthalpy of air-water vapor mixture at bulk water temperature and enthalpy of air-water vapor mixture at inlet wet-bulb temperature, respectively.

H. Pipe Sizing

In order to obtain a reasonable estimate for the pipe diameters, it was necessary to specify allowable pressure drops and maximum velocities for the fluids. Some guidelines for maximum allowable fluid flow characteristics included that "limits on steam or line should be 61 m/s (200 ft/s) and a pressure drop of 0.1 bar/100 m or 0.5 psi/100 ft of pipe." [http://www.cheresources.com/exprules.shtml (18)] In addition, the 1997 ASHRAE Handbook (19) recommended that water velocities for general service be between 4 and 10 ft/s.

For design purposes, reasonable pressure drops and maximum velocities were specified to be 100 Pa/m and 2 m/s, respectively, and a minimum pipe diameter was determined based upon these figures. The minimum pipe diameter based upon the maximum velocity is given by:

$$D = 2\sqrt{\frac{\dot{m}}{V\rho\pi}}. \tag{II-30}$$

Since the mass flow rate is given by

$$\dot{m} = \rho A V \tag{II-31}$$

and can be related to the diameter of the pipe via the cross-sectional area:

$$A = \pi (D/2)^2 = \frac{\dot{m}}{V\rho}. \tag{II-32}$$

To determine the minimum diameter based upon pressure losses in the pipes, the assumption will be made that there are no appreciable changes in height or velocity throughout the length of the pipe. Then, by rearranging the Bernoulli equation modified for major losses, we have the pressure drop per unit length given by:

$$\frac{\Delta p}{l} = \frac{g\rho}{l}\sum h_L = \rho\frac{fV^2}{2D} \qquad \text{(II-33)}$$

where f is the friction factor given by the Colebrook formula

$$\frac{1}{\sqrt{f}} = -2\log\left(\frac{\varepsilon/D}{3.7} + \frac{2.51}{\mathrm{Re}\sqrt{f}}\right) \qquad \text{(II-34)}$$

when in the turbulent regime and by

$$f = \frac{64}{\mathrm{Re}} \qquad \text{(II-35)}$$

when in the laminar region. However, this relationship cannot be solved directly for the diameter due to the implicit nature of the Colebrook formula. Instead, it is necessary to iterate on the solution until an appropriate diameter is found. These formulas may be found in any standard fluid mechanics text, such as that by Munson. A graphical relationship between the Reynold's number and friction factor is a chart known as the "Moody Diagram."

However, in our system there are significant pressure changes due to height due to the elevation of the turbine and condenser. This would change the energy balance of the system to reflect a pressure drop per unit length of pipe of:

$$\frac{\Delta p}{l} = \frac{\rho g}{l}\left(\Delta z + \sum h_l\right) = \rho g\frac{\Delta z}{l} + \rho\frac{fV^2}{2D} \qquad \text{(II-36)}$$

However, the change in height over the length of the pipe should be approximately equal to one, leading to the simplified equation:

$$\frac{\Delta p}{l} = \rho g + \rho\frac{fV^2}{2D} \qquad \text{(II-37)}$$

It is likely that the losses due to the height difference will be negligible compared to those due to friction in the pipes.

In addition, for the diameter of the pipe from the condenser to the cooling tower, the height analysis should give the appropriate height to handle any pipe size, regardless of pressure drops due to the length of the pipe.

For design purposes, the minimum pipe diameter calculated by this method was specified. For the equipment selection of pipes, it will be necessary to use the next largest size, since the casting of nonstandard pipe can be expensive. It is likely that the fluid flow rates will require large pipes that will have to be custom cast.

I. System Analysis

Once the different components of the salt gradient solar pond's power cycle have been analyzed, it is necessary to determine the efficiency of the overall system as the ratio of the net work of the system to the heat input to the system. The net work of the system is given by:

$$\begin{aligned} W_{net} &= [\text{Work output from system}] - [\text{Work input to system}] \\ &= [W_{turbine}] - [W_{pump} + W_{condenser} + W_{tower} + W_{vacuum}] \end{aligned} \tag{II-37}$$

The heat input to the system can be found by analyzing the solar pond, since the major heat input is that from solar radiation. The energy of the pond can be found from the enthalpy change across the pond, or

$$\begin{aligned} Q &= \sum \dot{m}_i h_i \\ &= \dot{m}_{separator} h_{separator} - \dot{m}_{pump} h_{pump} - \dot{m}_{tower} h_{tower} \\ &= \dot{m}_{separator} \left(h_{separator} - f h_{tower} - (1-f) h_{pump} \right) \end{aligned} \tag{II-38}$$

The first law efficiency is then given by

$$\eta = \frac{W_{net}}{Q} \tag{II-39}$$

J. Pond Data Estimation

In order to determine an approximate size for the solar pond, the average insolation, I, on Bakersfield, CA was considered to be 20 GJ/m^2/day (since this is an average value from Table I-1), with a pond capable of giving 20 % efficiency, η [Tabor (20)]. Since the total heat input to the pond, Q, was determined for the calculation of the first law efficiency of the power cycle, the pond area can be estimated from:

$$Area = \frac{Q}{\eta I} \tag{II-40}$$

The volume of water can then be found as the product of the pond area and depth. The salt needed can be estimated from the water requirements and the salinity of the pond at each layer.

The water losses from the pond due to evaporation can be estimated from the correlation presented in the ASHRAE handbook (19):

$$w_p = \frac{A(95 + .425\upsilon)}{h_{fg}}(p_w - p_a) \tag{II-41}$$

Where the units are as follows:

w_p = evaporation rate of water, lb/hr

A = area of pool surface, ft^2

υ = air velocity over pool, fpm

h_{fg} = latent heat of vaporization at dew point at water surface temperature, BTU/lb

p_a = saturation vapor pressure at dew point of ambient air, in. Hg

p_w = saturation vapor pressure at temperature of surface water, in. Hg

III. Design Results

A. Preliminary Design Analysis

In designing the power cycle for the salt-gradient solar pond, the salinity of the pond and the flash pressure had to be established.

The property data for saltwater given in Khan (8) is only valid up to 160 gm/kg (for specific heat and boiling point elevation), which is below the near saturation salinities of active solar ponds. For instance, the solar pond at El Paso, Texas has a salinity of 260 gm/kg in its lower convecting zone (21). The calculated effect of decreasing salinity on flash fraction is shown in Figure III-1. The figure clearly shows that decreasing salinity and flash pressure increases the flash fraction. However, physical constraints on the construction and maintenance of the separator prohibit low flash pressures. In fact, Penn Separator Co. does not make any separators that operate at sub-atmospheric pressures. Likewise, the solar pond needs to be operated at higher salinities in order to maintain its salinity gradient. However, since the property data was not available for the higher salinity values, a salinity of 150 gm/kg was chosen for the analysis of the cycle.

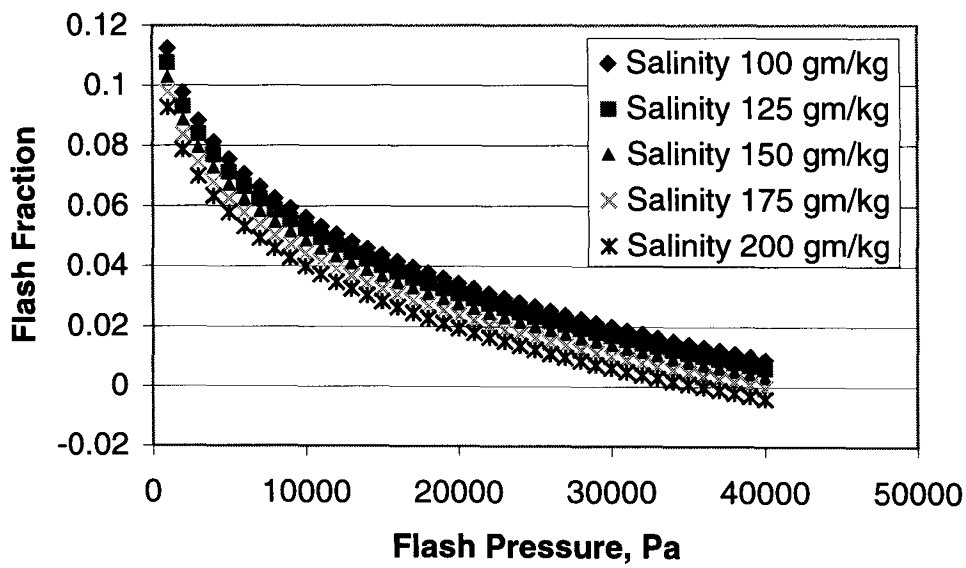


***Figure III-1*: The effect of salinity and flash pressure on the flash fraction in the separator.**

Once the salinity for the pond had been chosen, the appropriate flash pressure for the cycle was examined by plotting the effect of the flash pressure on the total power output of the system. Here, total power is defined as the power available from the turbine after the power to run the pump that returns the brine to the pond from the separator is subtracted. The results are shown in Figures III-2 and III-3. Clearly, the lower the flash pressure, the more power will be available. It must also be noted that the flash pressure cannot exceed 0.35 bar, as the separator will not produce steam above this pressure. A compromise flash pressure of 0.287 bar was chosen for the design cycle.

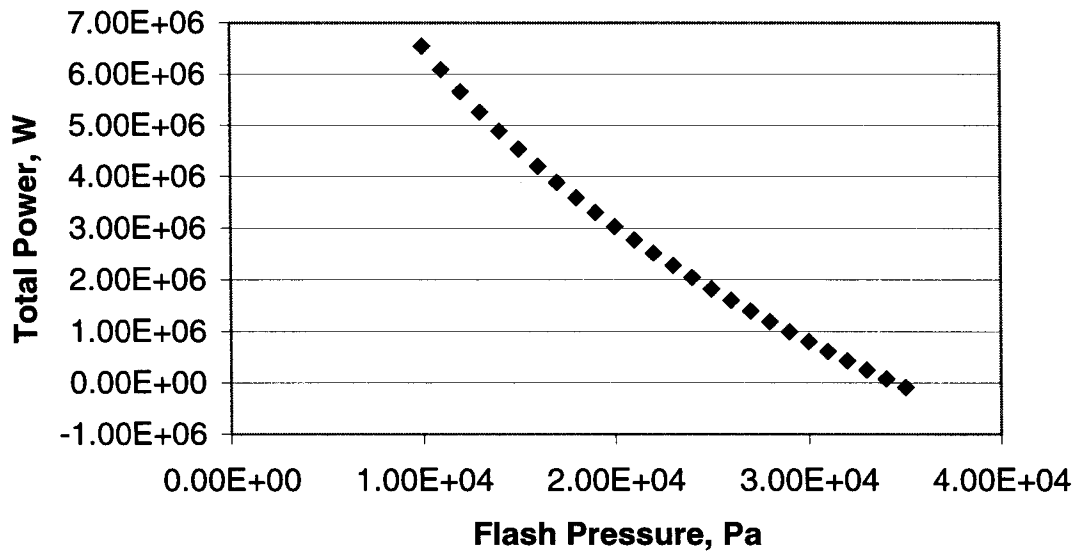


***Figure III-2*: The effect of flash pressure on the total available power of the system, per 1000 kg/s leaving the pond with brine salinity of 150 gm/kg.**

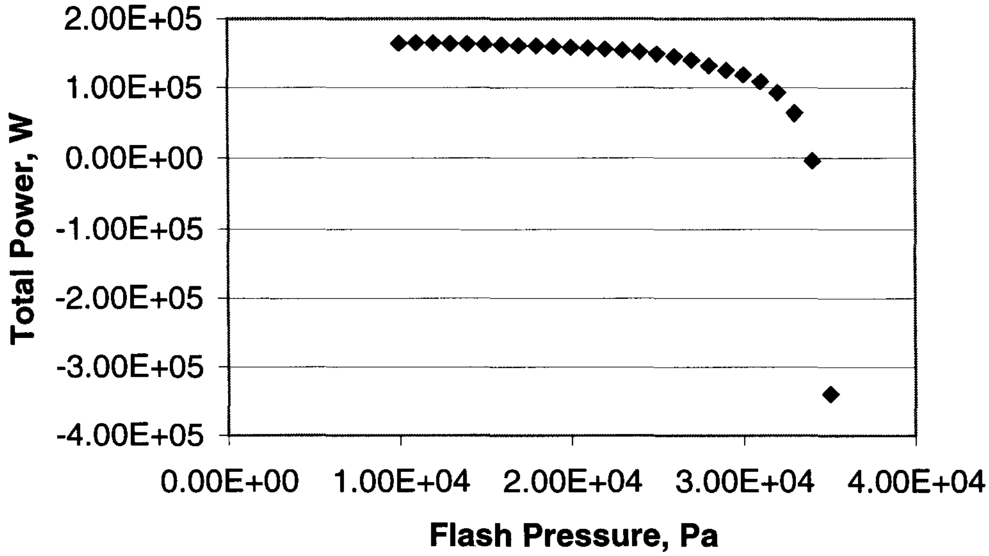


***Figure III-3*: The effect of flash pressure on the total available power, per 1 kg/s through turbine with brine salinity of 150 gm/kg.**

B. Code Results

This section of the report deals primarily with the results of the code. Equipment selection details will be given in the following section.

Once the flash pressure and salinity has been chosen, the MATLAB code developed for the analysis of the power cycle could be executed to determine the other design data for the cycle. Some of the code's input assumptions are given in Table III-1.

***Table III-1*: MATLAB code for assumptions for pond specifications.**

INPUT	
Temperature of pond bottom	353.150000 K
Pressure at pond bottom	147951.028487
Ambient Temperature	294.260000 K
Ambient Pressure	101300.000000 Pa
Salinity of lower convecting zone	150.000000 gm/kg
Depth of lower convecting zone	2.000000 m
Salinity of upper convecting zone	50.000000 gm/kg
Depth of upper convecting zone	0.500000 m
Total depth of pond	4.500000 m
Efficiency of turbine	0.800000
Efficiency of pump	0.800000
Mass flow of brine leaving pond	6000.000000 kg/s
Flash pressure	28700.000000 Pa

Table III-2 gives the code's calculations for the throttle and separator. A standard gate or ball valve can provide a throttle coefficient of 16.8 [Munson (11)]. The separator calculations showed that a horizontal separator would be the most economical for our situation, due to the excessive height of a vertical separator. The salinity of the brine after the flash process is only 1% more than the salinity of the bottom of the pond, and well below the 350 gm/kg required for precipitation of the salt.

***Table III-2*: Throttle and separator design specifications.**

Throttle calculations	
Diameter of pipe	1.877837 m
Pressure entering throttle	65000.000000 Pa
Throttle coefficient	16.755636
Height of throttle above pond	3.416032 m
Separator Calculations	
Flash fraction	0.007491
Salinity of brine after flashing	151.132156 gm/kg
Flash temperature	348.232944 K
Diameter of flash drum	14.077711 m
Height of Liquid	7.038856 m
Length of Vessel	42.233133 m

Table III-3 gives the pumping requirements to return the liquid brine to the pond after it exits the separator. Glancing at the table shows that the net positive suction head is given to be 5 m, which is the assumed depth of the pump below the separator. Since the liquid exiting the pump is at the vapor pressure, the only source of $NPSH_A$ is from the depth of the pump. However, with careful pump selection and deployment, it should be possible to work around this difficulty. For example, using 15 pumps provides a more reasonable flow rate for the available net positive suction head.

***Table III-3*: Pump specifications for returning brine from separator.**

Pump 1 Calculations	
Mass flow through pump	5955.054744 kg/s
Volumetric flow through pump	5.478051 m^3/s
Pumping power required	1711344.486056 W
Net positive suction head available	5.000000 m
Necessary head rise	23.435416 m
Pump 1 location with x pumps	
Number of pumps	15.000000
Mass flow per pump	397.003650 kg/s
Diameter of pipe	0.482178 m
Volumetric flow per pump	0.365203 m^3/s
Pumping power required, per pump	108808.598317 W
Net positive suction head available	5.000000 m
Necessary head rise per pump	22.350626 m

The turbine specifications are given in Table III-4. The additional power above the 5 MW peeking requirement for the cycle is to provide power for the mechanical vacuum, fans for the cooling tower, and pumps for the system.

***Table III-4*: Turbine specifications.**

Turbine Calculations	
Mass flow through turbine	44.945256 kg/s
Turbine inlet temperature	348.232944 K
Turbine inlet pressure	28700.000000 Pa
Pipe diameter to turbine	2.285125 m
Pipe diameter with height	2.285125 m
Quality of Isentropic Process	0.940000
Quality at turbine exit	0.956533
Temperature at turbine exit	317.880618 K
Pressure at turbine exit	9438.444719 Pa
Turbine power	7120859.301970 W

The condenser specifications are given in Table III-5. The length and diameter of the vessel seem to be of manageable proportions, however, the specified inlet diameter is nearly equal to the diameter of the vessel. It may be necessary to adjust the size of the vessel to accommodate the vapor inlet diameter, or to accept greater pressure drops in the inlet duct to allow a smaller diameter.

***Table III-5*: Condenser specifications.**

Condenser Calculations	
Temperature of vapor in	317.880618 K
Temperature of vapor out	312.880618 K
Temperature of liquid in	304.853864 K
Temperature of liquid out	314.853864 K
Condenser operating pressure	9438.444719 Pa
Allowable pressure drop	0.500000 Pa/m
Entrainment	0.050000
Mass flow rate of vapor in	44.945256 kg/s
Mass flow rate of vapor out	0.009357 kg/s
Mass flow rate of liquid in	1391.897153 kg/s
Mass flow rate of liquid out	1436.842409 kg/s
Number of nozzles	194.000000
Length of vessel	2.958902 m
Diameter of vessel	1.017534 m
Diameter of vapor inlet	0.943634 m
Volumetric Flow Rate of Vapor out	188.615731 cfm
Estimated Vacuum Power	14913.997432 W

Table III-6 gives the cooling tower requirements for the system. The 18 °F cooling range makes the cooling tower a medium range tower, and should be provided by commercial manufacturers.

Table III-6: **Cooling tower specifications.**

Cooling Tower Specifications	
Temperature of liquid in	314.853864 K, 107.066955 F
Temperature of liquid out	304.853864 K, 89.066955 F
Temperature of air in	310.400000 K, 99.050000 F
Temperature of air out	330.794302 K, 135.759744 F
Wet bulb temperature of inlet air	299.853864 K, 80.066955 F
Approach	5.000000 K, 9.000000 F
Cooling Range	10.000000 K, 18.000000 F
Mass flow rate of water	1392.761153 kg/s
Volumetric Flow Rate of Water	22437.998309 gal/min
Mass flow rate of air	2865.532446 kg/s
L/G	0.486039
KaV/L	0.960089
Estimated power requirement	335564.942212 W

The pumping requirements needed to raise water from the cooling tower to the condenser are given in Table III-7. Two standard centrifugal pump should meet these requirements.

Table III-7: **Requirements for pump to raise cooling water to condenser.**

Pump 2 Calculations	
Mass flow rate	1391.897153 kg/s
Volumetric flow rate	1.398709 m^3/s
Diameter of pipe	0.943634 m
Pumping power	52822.927887 W
Necessary head rise	3.094826 m
Net positive suction head available	9.897322 m
Pump 2 location with x pumps	
Number of pumps	2.000000
Mass flow per pump	695.948577 kg/s
Diameter of pipe	0.667250 m
Volumetric flow per pump	0.699354 m^3/s
Pumping power required, per pump	9461.687797 W
Net positive suction head available	9.897322 m
Necessary head rise per pump	1.108696 m

The available power for the system is just over the 5 MW specified. The estimated height of the turbine and condenser is around 12 m above the pond surface, which is far less than the original estimates. The data given on the pond in Table III-8 is based on rough estimations, and is intended to be used a starting point for PONDFEAS calculations.

***Table III-8*: Data on pond requirements and other information.**

Other Information	
Available power	5085428.459680 W
Height of turbine/condenser	10.005553 m
Pipe Diameter, t/c to ground	0.960491 m
Heat Input	110681718.908912 W
First law efficiency	0.045946
Estimated Insolation	20000000.000000 J/m^2-day
Estimated pond efficiency	0.200000
Estimated pond area	2390725.128433 m^2
Estimated pond volume	10758263.077946 m^3
Estimated pond mass	11198258941.003599 kg
Estimated salt mass	1550255061.368116 kg
Estimated evaporation losses	0.225200 kg/s
Fresh water generation per heat input	0.000000

C. Equipment Selection

Based on the results of the design analyses the overall system can be modeled. This system includes such specifications as the operational states and sizes of equipment used and the amount of utilities needed. The construction of a very large salt pond and the associated power pland require a large initial investment.

This initial investment has been broken down into investment costs, real estate costs, construction costs, gathering system costs, power plant costs, and piping costs. When possible, vendor quotes have been used to approximate equipment prices and installation costs.

Often vendor quotes have not been available. These instances, and the assumptions made in estimation techniques will be documented in the sections to follow. Often, these approximations and assumptions have been used based on published guidelines (22). These guidelines allow for the estimation of equipment costs based on size relative to a reference example according to the relationship:

$$C = C_r(S / S_r)^m \tag{III-1}$$

...where reference values and exponents have been compiled by researchers (Boehm see Burmeister).

However, these reference values were compiled in 1987, and results should be corrected for inflation. Using the Consumer Price Index (Burmeister), current price equivalents are approximated as:

$$\$_{now} = \$_{then} \times (CPI_{now} / CPI_{then}) \tag{III-2}$$

Installation costs are not included in these equipment approximations (they are however included in the piping approximation). Installation is approximated at 50% (22) of the equipment cost in these cases.

In addition to the operational states and sizes, and approximated costs of equipment selected, any applicable research, assumptions, or additional estimation techniques are also addressed in the following discussions.

1. Real Estate

The size of the pond was estimated as outlined in the design analyses section to be 2.5 million square meters. PONDFEAS confirmed that this size pond was appropriate for storing the requirement amount of thermal energy

The calculated pond area was 250 hectares, or about 620 acres, including some room for the power plant. Land costs near Bakersfield, CA are about $5000 per Hectare. Using this information, among the various other inputs, PONDFEAS estimates the cost of real estate at $2, 072, 955.

A caveat should be introduced here. . PONDFEAS is an independently supplied program, and the source code is unavailable. The assumptions and calculations that it uses to produce outputs are unknown, and thus the appropriateness of their application is uncontrolled. For this reason, the results gained from PONDFEAS have been regarded cautiously.

2. Pond Construction Costs

One of the most useful functions of PONDFEAS is the ability to approximate various aspects of pond construction costs. As has been outlined, various environmental and economic data is entered as inputs for the program. Based on information supplied about Bakersfield, the following construction costs are identified: Excavation: $13, 903, 534; Plastic Liner: $36, 837, 536; and Salt: $51, 050, 724.

In addition, water and manpower must be considered when addressing construction costs. The cost to fill the pond with water is based on an approximation of $250 per acre-feet (with an 8 ft. depth). The resulting cost for water for is approximated as $6, 250.

Manpower is approximated based on norms for medium scale construction projects. An estimate based on judgement suggests $2, 800, 000.

The same caveat discussed with the real estate costs applies here. The pond has lent itself to immensely expensive construction materials, and the results should be used very cautiously.

3. Gathering System

The flash separator must separate a large amount of fluid to obtain steam flow rates capable of driving the turbine. In order for the plant to operate safely, the separator must be large enough to store 10 minutes worth of feed flow from the pond in a shut-down situation. With over 90, 000 gal/ min of flow, the separator must be extremely large. The inlet to the separator should be at a height relative to the liquid level of the separator. Because of how tall the separator would have to be, additional pumping would be required to reach this inlet. As designed, the height of the inlet would be 33 meters above ground level. Thus, the separator used is horizontally oriented.

This separator is 40 meters long and 14 meters in diameter. Even with this orientation, the separator still must be partially buried (13 meters, infact) to avoid vaporization of pond water pumped to the inlet. Figure III-4 illustrates the relative heights that various system components will be installed at. This vessel is actually larger than any separators that were found commercially. In addition, none of the separator manufacturers contacted make separators designed to operate safely at sub-atmospheric pressures. The result is that this separator will have to be manufactured individually, and therefore will likely be quite expensive. Assuming such a vessel to be roughly three times the cost of a similarly dimensioned storage tank supporting a sub-atmospheric pressure, estimated cost is \$500, 000.

The brine feedback is costly. 86, 000 gal/min. of brine is not flashed and must be returned to the pond. As a result, both the initial investment and the operating costs of the pumps are extremely high.

Centrifugal pumps were chosen for this application because of their high efficiency, a must considering the already marginally feasible system they are supporting. Because the brine exits the separator already at vapor pressure, any available NPSH is the result of placing the pumps even further below ground level than the separator bottom. Figure III-4 illustrates this. A very large pump would have to be selected to accommodate the necessary flow rate. Such a pump would require far too much suction head to avoid cavitation. Several pumping schemes were considered before arriving at one that reasonably accommodated the needs of the system.

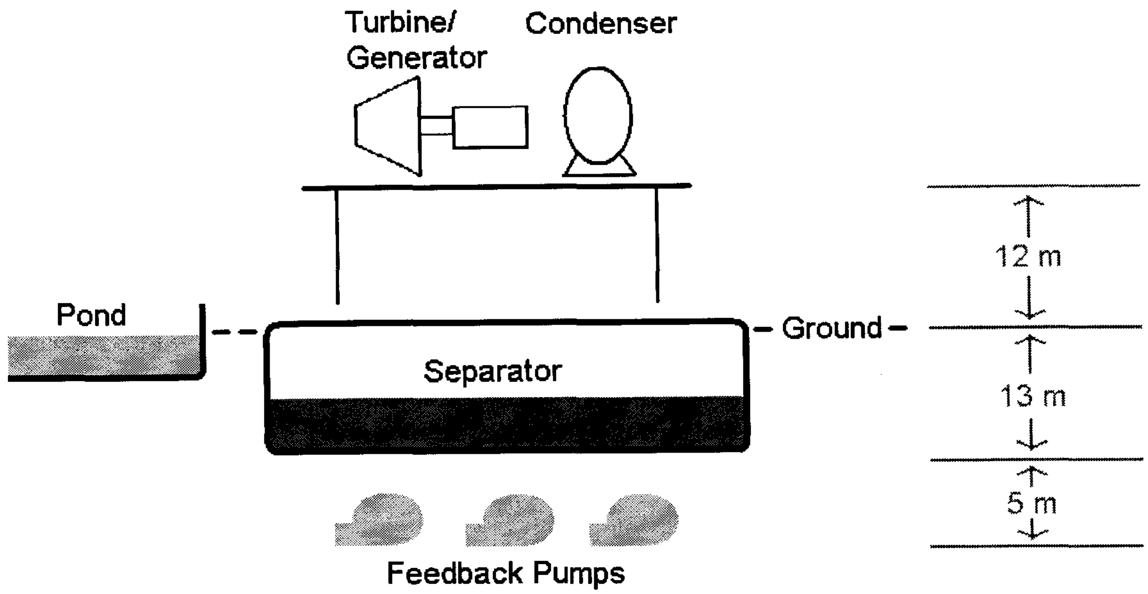


***Figure III-4*:** **The separator must be partially buried to avoid pumping to the inlet. The brine feedback pumps are further buried to create more $NPSH_A$. The turbine and condenser must be elevated to prevent vaporization between the condenser and the cooling tower.**

Goulds is the largest manufacturer of centrifugal pumps, and their line of pumps was selected for this application. 15 Model 3415 pumps operating at 1180 RPM, and dimensioned at 14x16-18 were selected. These pumps are buried an additional 20 feet below the bottom of the separator. Each of these pumps will handle about 5800 gal/ min. of brine against approximately 120 ft of head, while requiring just under 18 feet of suction head.

It is worth noting that the 17" diameter suction inlet on the pump is slightly smaller than the 19" limit recommended by the pipe sizing analysis to avoid excessively large pressure drops in the piping. This difference is small enough to defer to the experience of Goulds.

Vendor quotes were not available from Goulds at the time this was prepared. Based on the estimation guidelines already outlined, the pumps (with motors) used for brine return should cost approximately $11, 000 each and require 168 kW each from the generated power.

Because the turbine and condenser are 12 meters above the ground, the cooling water must be pumped back up to the condenser. Because the pressure in the condenser is sub-atmospheric, the pump system only needs to overcome less than 6 meters of head rise. The flow through the pump is 22, 200 gal/ min, and the $NPSH_A$ is nearly 10 meters.

With so much flow against such little head rise, several smaller pumps will need to be used. The system selected uses 2 pumps in parallel. Goulds offers a model 3498 pump that will raise 3 meters head and handle flow rates of 2200 gal/min. These pumps need to operated at 1180 RPM and dimensioned at 20x20-18. They only require just over 9 kW of power each.

Again, as no quotes were obtained from Goulds, the cost estimation is based on the discussed guidelines. Each of the cooling water pumps will cost approximately $5000.

4. Power Plant Costs

The turbine will operate with a flow rate of 1450 gal/ min and a 67 % pressure drop. At peak operation it will produce 7.1 MW of power, which is enough to power the parasitic processes and still meet the 5MW specification.

General Electric proved extremely hesitant to provide any information at all to non-G.E-certified customers. As a consequence, very little is know about the operation of the turbine beyond what has been specified by system requirements. Using the outlined const-estimation techniques, the turbine is estimated at $139, 000. Likewise, the generator will cost $35, 000.

The condenser must operate at a sub-atmospheric pressure of 9400 Pa, and accommodate 45 kg/s of steam at an inlet temperature of 45°C. The maximum allowable pressure drop is .5 Pa/m.

A direct contact spray condenser was selected to condense the vapor from the turbine for the ease of design and higher efficiency. The as-designed vessel is .4 meters long and just over 1 meter in diameter. 194 spray nozzles are used. Using the discussed approximation guidelines, the condenser should cost about $161,000.

A necessary aspect of running the condenser is evacuating the non-condensable vapor and maintaining vacuum pressure. Ejectors can easily perform this function without requiring very much maintenance, but our design work in that direction indicated that a two-stage ejector would actually require more steam than was being used for the turbine.

Instead, a liquid ring mechanical vacuum pump will be used. This pump needs to maintain a sub-atmospheric pressure in the condenser of 9400 Pa. The flow rate of the vapor it needs to evacuate is 188 cfm.

Stokes Vacuum Inc. manufactures liquid ring vacuum pumps. The selected pump is a Stokes CHR series single stage liquid ring pump, model 1200 operating at 1740 RPM. This pump only requires 15 kW of power. It does require 13.7 gal/ min service water, but this is only a fraction of the cooling water already being pumped back up to the condenser.

A vendor quote was not obtained from Stokes vacuum, but based on similar pumping applications, this pump is estimated to cost $10, 000.

The flow rate of water through the cooling tower is 1440 kg/s. The inlet temperature is 41°C, and should be reduced to 31°C at the exit.

An induced draft, counter-flow cooling tower was selected for this application. Marley Cooling Tower supplied a quote for a cooling tower that would operate at these conditions. It is a tower model W478-4.0-3 with 3 cells, with a 28 ft. fan diameter. It requires 3 motors using 112 MW of power each. The tower is 13 meters wide, 44 meters long, and 13 meters tall. The quote they supplied includes installation and is $588, 500.

5. Piping Costs

Estimating the amount of piping needed to support this system is an inexact exercise. Only very detailed three-dimensional modeling of the plant layout would yield accurate accounting of needed pipe lengths and fittings. What is left to work with is the most manageable approximation based on the sizes of the equipment used.

With the amount and size of process equipment used, approximately 1200 meters (4000 ft) of pipe of varying diameter should be purchased and on hand for construction. Corrected for inflation since publishing (although mitigated with common since), rule of thumb guidelines (23) seem indicate that pipe may cost an average of $50 per foot (including insulation, fitting, valves, and installation) for both the above and below ground applications we have specified. This leads to a very rough approximation of $200, 000 for the cost of piping for the entire plant.

IV. Economic Analysis Procedure

Determining the economic feasibility of a power generation cycle utilizing a salt-gradient solar pond is the purpose of this economic analysis. In developing this design, the TEAM constantly considered alternatives, focused on the differences between those alternatives, used a consistent viewpoint as a borrower, considered all relevant criteria on which the decision was based, made uncertainties explicit, and reviewed all decisions. (24). The TEAM created an Excel spreadsheet in order to calculate the necessary economic considerations. The economic analysis involves the systematic evaluation of the costs and benefits of this proposed pond and power generation cycle.

Understanding the objective of the facility enabled TEAM to evaluate its viability. The objective of facility investment in the private sector is the profit maximization within a specific time frame. Given this objective, the method of economic analysis will be judged by the reliability and ease with which a correct conclusion may be reached about the designed system.

Using guidelines from The Engineer's Cost Handbook: Tools for Managing Project Costs by Richard E. Westney, P.E., (25) a systematic approach for economic evaluation of our facility consists of the following major steps:

1. Generate a project for investment consideration.
2. Establish the planning horizon for economic analysis.
3. Estimate the cash flow profile for the project.
4. Specify the minimum attractive rate of return (MARR).
5. Determine the present worth of the project.
6. Establish the criterion for accepting or rejecting a proposal on the basis of the objective of the investment.
7. Accept or reject a proposal on the basis of the established criterion.

The project has already been outlined in the previous pages of this report. Now a plan for implementing the proposal needs to be developed.

TEAM developed a planning period for the project. The planning horizon is the period of time that a company looks ahead in regards to an investment. The ability to forecast a project with some degree of accuracy limits the period of time selected. For the initial investment, the selection of the planning horizon is influenced by the useful life of a facility, since the disposal of usable assets generally involves suffering financial losses. For the power plant that TEAM designed, the useful life of the plant was assumed to be 30 years.

Next, TEAM calculated the cash flow profile of the project. In economic evaluations, project alternatives are represented by their cash flow profiles over the n years or periods in the planning horizon. Thus, the interest periods are normally assumed to be in years (t = 0, 1, 2 ... n with t = 0 representing the present time). Let B_t be the annual income at the end of year t for an investment project. Let C_t be the annual cost at the end of year t for the same investment project. The net annual cash flow is defined as the annual benefit in excess of the annual cost, and is denoted by A_t at the end of year t for an investment project. Then, for year t:

$$A_t = B_t - C_t \qquad \text{(IV-1)}$$

where A_t is positive, negative or zero depends on the values of B_t and C_t.

To obtain an accurate estimate of costs in the cash flow profile for the installation and operation of a project, it is necessary to specify the resources required to construct and operate the proposed physical facility. Typically, each of the labor and material resources required by the facility is multiplied by its price, and the products are then summed to obtain the total costs. The initial investment required for implementing the salt-gradient solar pond and a power plant was calculated by considering the real estate costs, construction costs, gathering system costs, power plant costs, and piping costs. The following equation shows the initial investment costs:

$$\begin{aligned}[\text{Initial Investment}] = {} & [\text{Land Costs}] + [\text{Construction Costs}] + \\ & [\text{Piping Costs}] + [\text{Gathering System Costs}] + \\ & [\text{Power Plant Costs}]\end{aligned} \qquad \text{(IV-2)}$$

Each of the costs considered in the initial investment can be broken down into more specific costs. The land costs only encompass the actual combined area that the solar pond and power plant require for installation. This relationship is shown as:

$$[\text{Land Costs}] = [\text{Area of Land (Pond and Power Plant)}] \times [\text{Cost per Unit Area}] \qquad \text{(IV-3)}$$

The next consideration was the construction of the solar pond on that land. Such a consideration involves the excavation costs, plastic liner costs, salt costs, water costs, manpower costs, wave control, control cost, fence cost, and engineering design cost. Each of these costs was determined using a feasibility program called PONDFEAS. This program enabled the TEAM to set the variable parameters such as pond location, climatic conditions, load type and schedule, type of conventional fuels, fuel costs, economic data, system cost figures, and system maintenance and repair costs. Using the MATLAB output from the design of the power plant, the TEAM was able to adjust each of these parameters. As Gerald Cler, author of PONDFEAS, stated in the user's manual:

> The excavation volume was calculated by assuming the pond is partially below the original grade. This volume was used to construct the berms that rise above grade, and the added fill required for the evaporation surface, minimizing the need to have fill brought to the site. All calculations performed in PONDFEAS assumed that the pond was a square since only the area was known and the lengths of the sides were not. The quantity of liner material was calculated to cover the entire pond, one meter of freeboard, material required for anchoring the liner to the top of the berm, and the evaporation surface. The quantity of salt required was calculated based on the area of the pond and the depth of each zone within the pond (PONDFEAS, 5).

The next calculations involved estimating the costs of the equipment used in the power plant. Many vendors were contacted to obtain quotes. These quotes were compared to the cost estimates calculated using the Boehm cost equations provided in Elements of Thermal-Fluid System Design by Louis C. Burmeister (22). The following relation was used to estimate the cost of condenser, pumps, and turbine/generator unit:

$$C = C_r (S / S_r)^m \qquad \text{(IV-4)}$$

where C was the cost, Cr was the reference cost, S was the size, Sr was the reference size, and m was an exponential factor. Some of the equipment required correction factors if the designed size did not fall within the acceptable size range.

The costs were in terms of 1987 US dollars, so a correction factor was used to determine the price in 2001 US dollars. The correction factor was calculated by reading the consumer price index, CPI, for 1987 and interpolating the CPI for 2001. In 1987, the CPI was 130. To obtain the CPI for 2001, the inflation rate was assumed to be 4.25% per year and the value was assumed to lie on the best-fit line relating previous consumer price indices. The CPI for 2001 is 190. Using the following relation, the correction factor that accounted for the inflation rate was used to estimate the cost of the equipment in 2001 US dollars:

$$C_{2001} = C_{1987} \frac{190}{130} \qquad \text{(IV-5)}$$

The variables used in estimating equipment using this method are shown in the following table.

Table IV-1. **Method of Cost Estimation [*Elements of Thermal-Fluid System Design(22)*]**

Equipment	m	Reference Cost Cr	Reference Size Sr	Size S	C ($)	Correction Factors	Cost (1987 $)	Cost (2001 $)
condenser	0.55	3000	10	7.00E+03	110,134.95		110,134.95	160,966.47
Pump1	0.58	7500	100	96.82119	7,360.78	*15 pumps	110,411.77	161,371.05
Pump2	0.58	7500	100	25.9	3,425.90	*2 pumps	6,851.81	10,014.18
Turbine	0.68	25000	1000	7120.859	94,986.53		94,986.53	138,826.47
Generator	0.95	3700	1000	7120.859	23,884.02		23,884.02	34,907.42

The reason we used this method of determining some costs due to the lack of response from vendors on equipment cost estimates. However, the TEAM was able to get a quote from Marley Cooling Towers for an estimate on the cooling tower itself. Given the desired specifications for the cooling tower design, Marley Cooling Towers delivered an estimate that seemed reasonably priced. The actual quote can be found in the appendix.

Unfortunately, the design specifications for the separator and for the mechanical vacuum pump limited the availability of an accurate estimate from any vendor. In fact, due to the low pressure requirements for the separator, one vendor informed the TEAM that no separator could even be built to support this design requirement. The costs of the separator and the mechanical vacuum pump were therefore estimated as $500,000 and $5,000, respectively. These estimates were derived from researching the average prices of separators and mechanical vacuum pumps available in industry. Piping costs were estimated using the mechanical engineering handbook, *Modern Cost Engineering Techniques(23).* The values were based on price per unit length multiplied by the length of the pipe.

The next costs that concerned the TEAM were installation costs. Each piece of equipment had an installation cost that was determined by multiplying a factor by the estimated cost of the equipment. The installation factor for the condenser, pumps, separator, cooling tower and the turbine is 0.25. The installation costs of the pond were included in the PONDFEAS analysis and were not accounted for using the same installation factor approach. A summary of the initial investment costs is shown in Table IV-4.

Table IV-2. **Initial Investment**

Land Costs (@ $5000/hectare)					Real Estate Costs
$ 2,072,955					$ 2,072,955.00
Excavation Costs (@ $2/m^3)	Plastic Liner Costs (@ $9/m^2)	Salt Costs (@ $25/ton)	Water Costs (@ $250/acre-ft)	Manpower Costs	
$13,903,534	$36,837,536.00	$ 51,050,724	$ 6,200.00	$ 1,000.00	
Wave Control (@ $1/m^2)	Control Cost ($2500)	Fence Cost (@ $20/m)	Engineering Design Cost		Construction Costs
$2,500,000	$ 2,500.00	$ 167,620.00	$11,406,487		$ 115,875,601
Pump Costs	Flash Separator Costs	Installation Costs			Gathering System Costs
$181,385.23	$ 500,000.00	$ 340,692.61			$ 1,022,077.84
Turbine Costs	Condenser Costs	Cooling Tower Costs	Installation Costs		Power Plant Costs
$ 73,733.89	$ 160,966.47	$ 588,500.00	$ 357,841.28		$ 1,281,041.64
Pipe Costs	Fitting Costs	Insulation Costs	Valve Costs	Installation Costs	Piping Costs
Estimated $40 per meter of pipe for piping costs--->					$ 200,000.00
					Initial Investment = $ 120,451,675

After the initial investment cost was determined, the annual costs were calculated. The operation and maintenance cost was calculated using the annual production of electricity multiplied by the operation and maintenance cost rate of 2.5 cents per kilowatt-hour. The make-up water cost was determined to average \$200,000 per year. The following equations outline the method for calculating annual costs. See Table IV-3 for a summary of the annual costs.

$$\begin{aligned}[\text{Annual Operation \& Maintenence Costs}] = [\text{Annual kWh Production}] \times \\ [\text{O \& M Cost Rate \$/kWh}]\end{aligned} \quad \text{(IV-6)}$$

$$[\text{Annual Costs}] = [\text{Annual O \& M Costs}] + [\text{Make - up Water Cost}] \quad \text{(IV-7)}$$

***Table IV-3*: Annual Costs**

Annual kWh Production	O & M Cost Rate $/kWh)	Electricity Produced	Annual O & M Cost
30,426,400.00	0.025	5.21E+03	$ 760,660.00
Make-up Water Costs			Make-up Water Costs
$ 200,000.00			$ 200,000.00
		Annual Costs =	$ 960,660.00

The annual receipts were then calculated using the annual production of electricity multiplied by the selling price per kilowatt-hour. The selling price was varied until an acceptable rate was found to be 12 cents per kilowatt-hour. See Table IV-4 below for the annual receipts. Equation IV-7 shows this relationship:

$$[\text{Annual Receipts}] = [\text{Annual kWh Production}] \times [\text{Selling Price per kWh}] \quad \text{(IV-8)}$$

***Table IV-4*: Annual Receipts**

Annual kWh Production	Selling Price per kWh	Annual Receipts
30,426,400	0.12	$ 3,651,168.00
	Annual Receipts =	$ 3,651,168.00

Depreciation for the power plant equipment and for the pond was the next item considered. Using the modified accelerated cost recovery system (MACRS) method of depreciation for the power plant equipment and assuming a recovery period of 20 years from the initial investment on the equipment, the depreciation of the equipment varied accordingly. A summary of the annual depreciation of equipment is shown here:

Table IV-5: **Annual Depreciation on Equipment**
(Using the Modified Accelerated Cost Recovery System)

MACRS Depreciation Rate	Year	Depreciation
0.0375	2002	$ 76,513
0.0722	2003	147,313
0.0688	2004	140,376
0.0618	2005	126,094
0.0571	2006	116,504
0.0528	2007	107,730
0.0489	2008	99,773
0.0452	2009	92,224
0.0447	2010	91,204
0.0447	2011	91,204
0.0446	2012	91,000
0.0446	2013	91,000
0.0446	2014	91,000
0.0446	2015	91,000
0.0446	2016	91,000
0.0446	2017	91,000
0.0446	2018	91,000
0.0446	2019	91,000
0.0446	2020	91,000
0.0446	2021	91,000
0.0223	2022	45,500
0	2023	0
0	2024	0
0	2025	0
0	2026	0
0	2027	0
0	2028	0
0	2029	0
0	2030	0
0	2031	0

For private corporations, the cash flow profile of a project is affected by the amount of taxation. In the context of tax liability, depreciation is the amount allowed as a deduction due to capital expenses in computing taxable income and, hence, income tax in any year. Thus, depreciation results in a reduction in tax liabilities. It is important to differentiate between the estimated useful life used in depreciation computations and the actual useful life of a facility. The former is often an arbitrary length of time, specified in the regulations of the U.S. Internal Revenue Service or a comparable organization. The depreciation allowance is a bookkeeping entry that does not involve an outlay of cash, but represents a systematic allocation of the cost of a physical facility over time.

When calculating the annual depreciation for the pond, the straight-line depreciation method was utilized. This method assumes the annual depreciation is fixed over the initial investment recovery period of 20 years for the pond. For full recovery of the initial investment cost, the annual depreciation had to be 5% of the initial cost of the pond. Since the pond costs \$ 117,948,556 originally, the annual depreciation of the pond is \$ 5,897,068 or simply:

$$[\text{Annual Pond Depreciation}] = [\text{Initial Investment of Pond}] \times [5\%] \qquad \text{(IV-9)}$$

The annual gross income of the project was calculated using the following relationship:

$$\begin{aligned}[\text{Gross Income}] = {} & [\text{Annual Receipts}] - [\text{Annual Costs}] \\ & - [\text{Depreciation of Equipment}] - [\text{Depreciation of Pond}]\end{aligned} \qquad \text{(IV-10)}$$

Over the 30-year life of the pond, the gross income was predicted for each year and is represented in Table IV-6.

***Table IV-6*: Annual Gross Income**

Year	Gross Income ($)
2002	-3,283,073
2003	-3,353,873
2004	-3,346,936
2005	-3,332,653
2006	-3,323,064
2007	-3,314,290
2008	-3,306,333
2009	-3,298,784
2010	-3,297,763
2011	-3,297,763
2012	-3,297,559
2013	-3,297,559
2014	-3,297,559
2015	-3,297,559
2016	-3,297,559
2017	-3,297,559
2018	-3,297,559
2019	-3,297,559
2020	-3,297,559
2021	-3,297,559
2022	-3,252,060
2023	2,690,508
2024	2,690,508
2025	2,690,508
2026	2,690,508
2027	2,690,508
2028	2,690,508
2029	2,690,508
2030	2,690,508
2031	2,690,508

Net income was determined for the 30-year period depending on the gross income. For the first 20 years of the life of the system, the gross income is negative. During these years, no taxes are applied to calculate the net income; that is, the net income equals the gross income. However, once the project has recovered from the initial costs of the equipment and the pond, a positive gross income will occur. During the final ten years of the life of the power plant and pond, state and federal taxes will be assessed accordingly. Both taxes depend on the gross income. Net income was calculated by:

$$[\text{Net Income}] = [\text{Gross Income}] - [\text{Federal Tax}] - [\text{State Tax}] \qquad \text{(IV-11)}$$

For the 30-year life of the facility, the net income was summarized in the following table.

***Table IV-7*: Annual Net Income**

Year	Federal Tax ($)	State Tax ($)	Net Income ($)
2002	0	0	-3,283,073
2003	0	0	-3,353,873
2004	0	0	-3,346,936
2005	0	0	-3,332,653
2006	0	0	-3,323,064
2007	0	0	-3,314,290
2008	0	0	-3,306,333
2009	0	0	-3,298,784
2010	0	0	-3,297,763
2011	0	0	-3,297,763
2012	0	0	-3,297,559
2013	0	0	-3,297,559
2014	0	0	-3,297,559
2015	0	0	-3,297,559
2016	0	0	-3,297,559
2017	0	0	-3,297,559
2018	0	0	-3,297,559
2019	0	0	-3,297,559
2020	0	0	-3,297,559
2021	0	0	-3,297,559
2022	0	0	-3,252,060
2023	914,773	53,810	1,721,925
2024	914,773	53,810	1,721,925
2025	914,773	53,810	1,721,925
2026	914,773	53,810	1,721,925
2027	914,773	53,810	1,721,925
2028	914,773	53,810	1,721,925
2029	914,773	53,810	1,721,925
2030	914,773	53,810	1,721,925
2031	914,773	53,810	1,721,925

The following figure shows the total adjusted income for the company over the 30-year life of the facility.

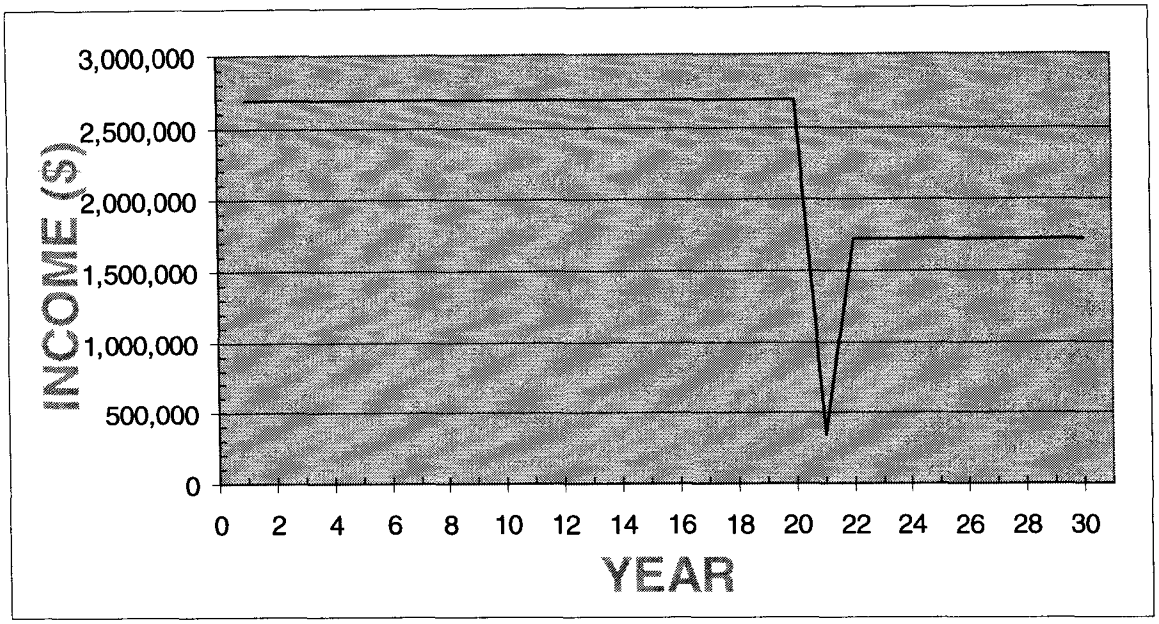


Figure IV-1. **Company Income Over the 30-year Life of the System**

Constructed facilities are inherently long-term investments with a deferred pay-off. The cost of capital or minimum attractive rate of return (MARR) depends on the real interest rate (i.e., market interest rate less the inflation rate) over the period of investment. As the cost of capital rises, it becomes less and less attractive to invest in a large facility because of the opportunities foregone over a long period of time. The real interest rate is calculated as the market interest rate less the general rate of inflation. Interest charges and the ultimate cost of projects are hard to predict due to the volatility of the market interest rate. In economic evaluation, a constant value of MARR over the planning horizon is often used to simplify the calculations. The use of a constant value for MARR is justified on the ground of long-term average of the cost of capital over the period of investment.

The management of TEAM decided to invest in the proposed project only if it will yield a return at least equal to the MARR. The MARR specified for the economic evaluation of our investment proposal was critically important in determining whether the project was worthwhile. Using several values of the MARR to assess the potential of the project allowed the TEAM to determine its best economic choice.

The present worth of the system was determined by the following equation:

$$[\text{Present Worth}] = [\text{PW}] = \sum_{t=0}^{n} \frac{C_r}{(1+d)^t} \qquad \text{(IV-12)}$$

Paying for the system was the next issue that faced the TEAM. After discovering that the State of California has a Buydown Program for renewable energy resources, the best option became obvious. In October 1999, the California Energy Commission released A Guidebook for Renewable Technology Program, Volume 3: Emerging Renewable Resources Account (26). Within that literature, the Buydown Program was summarized. The State of California applies 10% of its Renewable Resource Trust Fund to Emerging Renewable Resources Account. Solar Thermal Electric energy falls under the category of emerging renewable resources. The Buydown Program is intended to reduce the net cost to the end user of generating systems and to stimulate substantial sales of such systems. The Buydown Program is designed to cover up to 50% of the total cost of the generating system. For this project, the Buydown Program needs to cover 50% in order for the project to be economically feasible.

Implementing the Buydown Program in our economic analysis, the net present worth was then calculated by subtracting 50% of the initial investment cost of the system from the present worth. The net present worth is calculated using the following equation:

$$[\text{Net Present Worth}] = [\text{NPW}] = \sum_{t=0}^{n} \frac{(B_t - C_t)}{(1+d)^t} \qquad \text{(IV-13)}$$

The net present worth for the system at various MARR rates is graphed in the Figure IV-2.

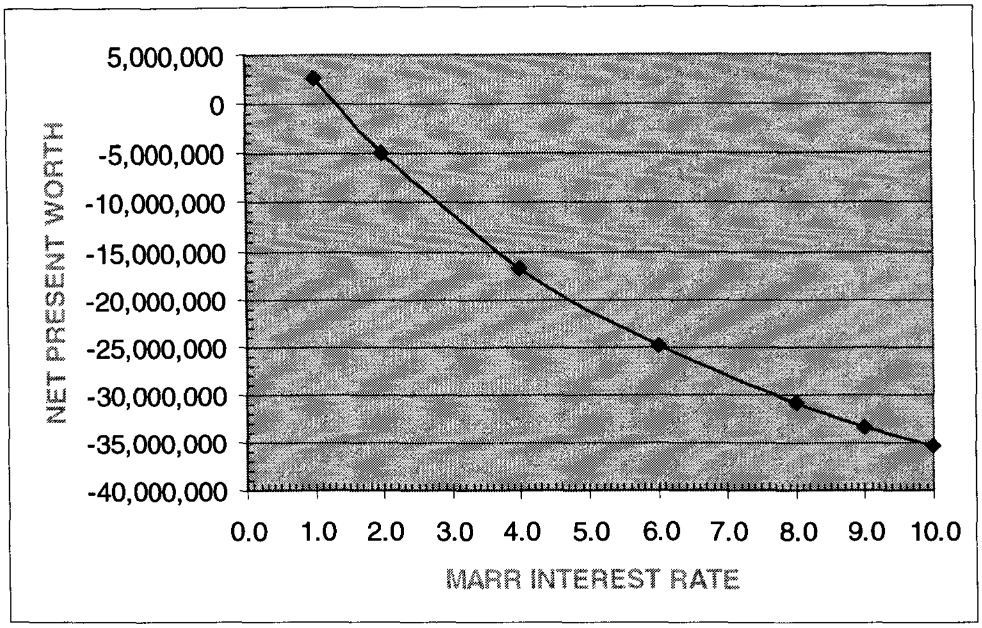


Figure IV-2. **The Net Present Worth versus the MARR Interest Rate**

The point where the curve crosses a net present worth value equal to zero represents the optimal MARR interest rate to apply in order to receive the minimum acceptable rate of return on the initial investments. This point corresponds to a MARR interest rate of approximately 1.3%.

Overall, the economic analysis proves the profitability of the project. This analysis enables TEAM to make a decision on implementing the designed system.

V. Future Research Opportunities

The system design outlined by this project did not explore every design parameter. Notably, little design work was actually performed in designing the pond itself. For instance, the brine temperature of 80°C was assumed as a conservative estimate of the temperature of the lower convective zone based on research into previous pond experience. This discussion is included as an example of possible improvements outside the scope of our design objective that could benefit the overall system design.

Although our results show an acceptable efficiency to produce power, improvements could be made to the cycle by increasing the pond (and thus the feed) temperature or lowering the flash pressure. These changes would increase the efficiency of the flash separation process, and result in improved performance for the entire cycle.

Figure V-1 illustrates the effect of changing the flash pressure on the cycle efficiency and the required feed flow rate in order to generate 5 MW of power. The decrease in the feed flow rate would make the cycle more feasible by decreasing the overall size of the separator and reducing concerns about the mass transfer of water into and out of the pond affecting the system stability and salinity gradient. The increase in cycle efficiency is also a favorable result, as it leads to a decrease in the overall size of the pond.

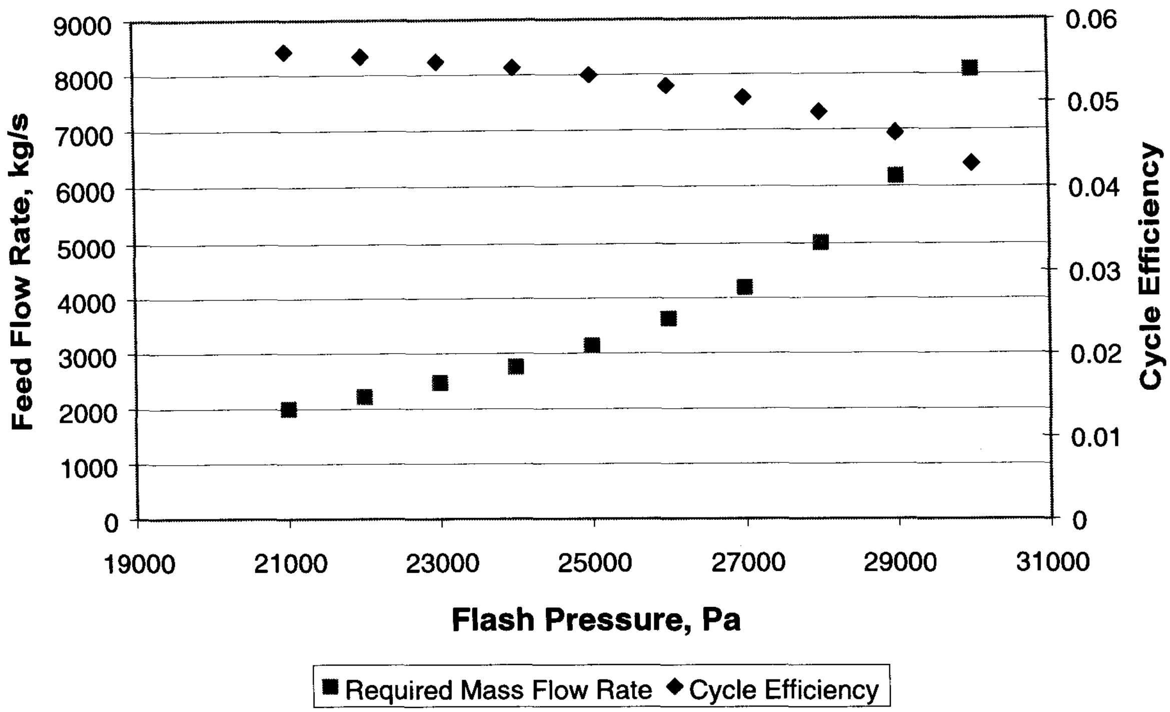


Figure V-1**: System Data as a function of flash pressure, assuming a pond salinity of 150 gm/kg and a Brine Temperature of 80 °C. Available power from the system is approximately 5 MW.**

Raising the feed temperature to the separator could be accomplished by raising the temperature of the lower convecting zone of the pond. The higher temperature water would have a higher flash fraction, as shown in Figure V-2, which would lead to a decrease in the feed flow rate, as shown in Figure V-3, and a decrease in the overall size of the separator. The system efficiency also increases with increased feed temperature, as shown in Figure V-3, which would lead to a smaller pond size, since less solar energy would be needed for the same amount of electricity produced.

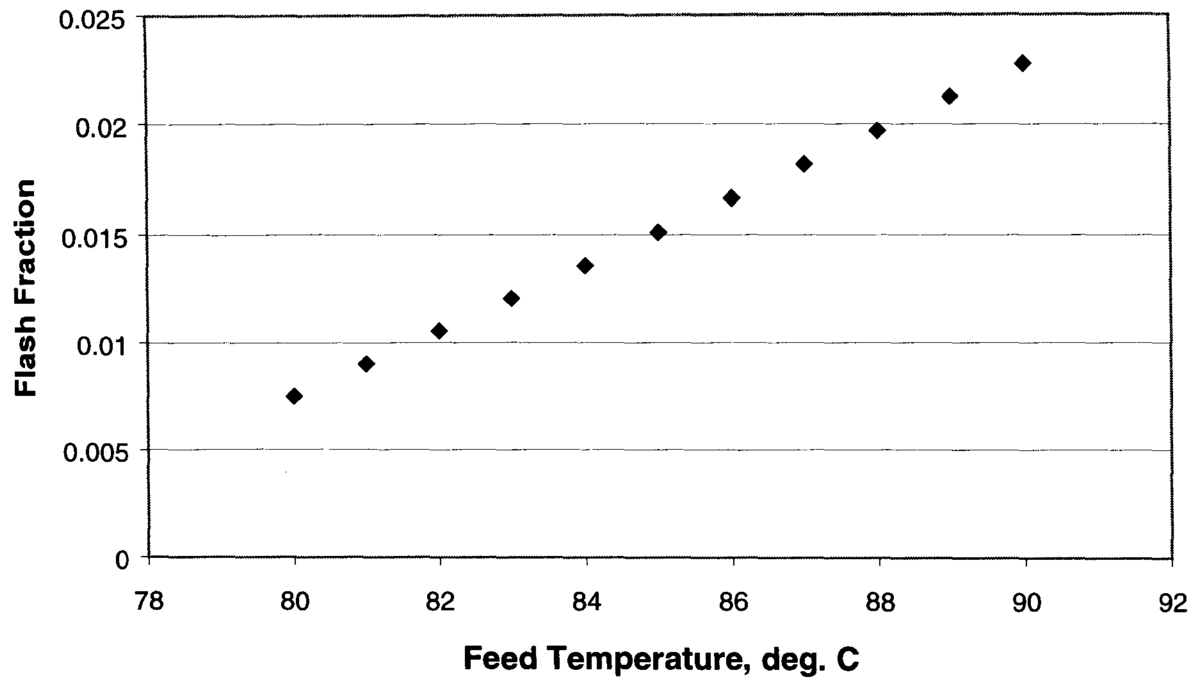


Figure V-2: **Flash fraction as a function of feed (pond) temperature at a salinity of 150 gm/kg.**

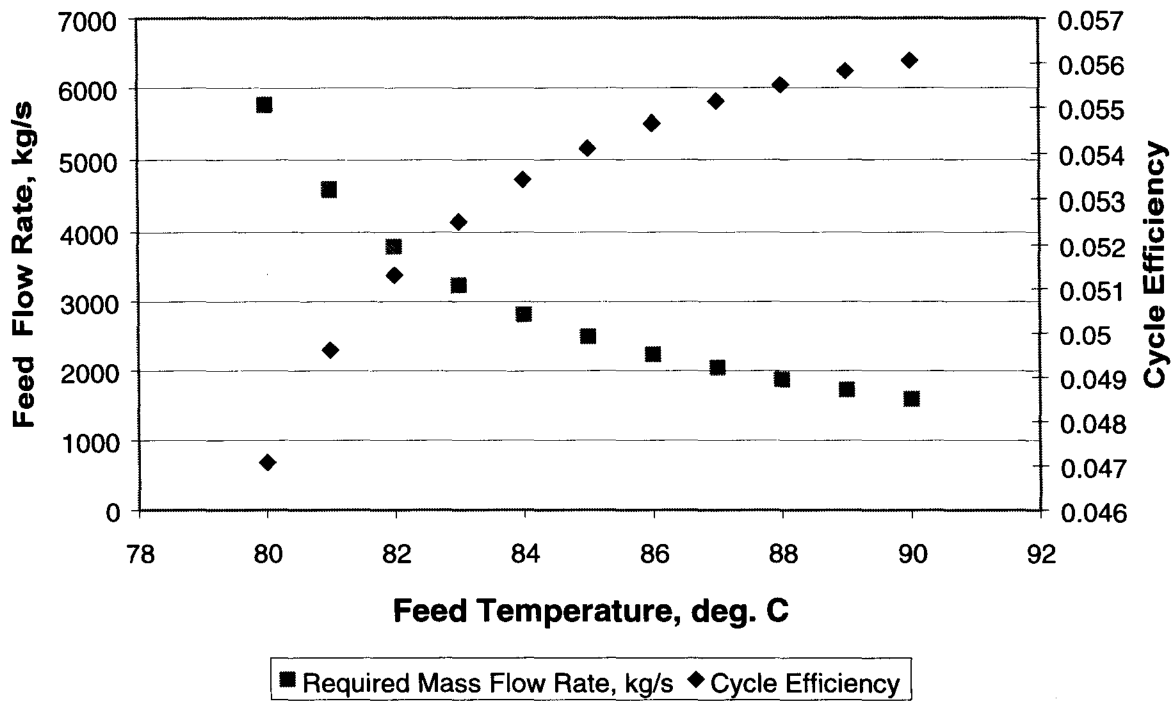


***Figure V-3*: System characteristics as a function of feed (pond) temperature, assuming a salinity of 150 gm/kg and a flash pressure 28700 Pa. Available power from the system is approximately 5 MW.**

Although increasing the brine temperature or decreasing the flash pressure would help the cycle's feasibility, it may not be physically or economically possible. For instance the separator would have to be redesigned to withstand very low pressures, which may require expensive manufacturing techniques. The pond would also have to be adjusted in order to produce higher temperatures without boiling which would destroy the salinity gradient.

VI. Conclusions and Recommendations

By selling electricity at the rate of twelve cents per kilowatt-hour, TEAM concludes that the design of a power generation cycle utilizing a salt gradient solar pond is possible. This price is roughly twice the current electricity rate in the state of California. Although this price is not competitive with the current market price, the proposed design does present a feasible alternative energy source.

An alternative to selling the power to the public at a price per kilowatt-hour would be to present the design to an industry for use solely in its plant operations as back-up power. This would provide the industry a reliable power source during the current California energy crisis. The industry would rely on the power plant during periods of rolling blackouts. Such a luxury would enable an industry to bypass the inconveniences resulting from lack of power to their own operations.

Before the construction of the power plant begins, the TEAM must conduct more research in the area of the separator design. In the design process, it became evident that a small flash fraction would require very low pressure and would limit the availability of a separator capable of withstanding such conditions. If it were possible to find a manufacturer of the separator that TEAM has designed, the power system could be built. Other limitations surrounding the flash cycle included generating the flow rate needed to flash enough steam to operate the turbine at maximum output.

The final recommendation from TEAM is to continue research efforts until a working system, complete with sized equipment, is readily available. This would require concentrated analysis of the separator in order to meet the design specifications.

Bibliography

1. http://www.dieoff.org/page84.htm

2. http://www.greentie.org/class/ixb08.htm

3. http://www.ci.bakerfield.ca.us/cityservices/water/indes.htm

4. http://www.bestplaces.net

5. Cler, Gerald L. and Ty Newell. PONDFEAS: A Feasibility Study and Design Tool for Salt Gradient Solar Ponds. Version FY90, September 1990. (Including pre-programmed inputs for solar data for various locations.)

6. Kornhauser. "Open Cycles for Power Generation from Solar Ponds." Presented at FORUM 2001: Washington, D. C., April, 2001

7. Krane, R. J. Class Lecture. University of Tennessee. February 15, 2001.

8. Khan, A. H. *Desalination Processes and Multistage Flash Distillation Practice.* Elsevier, Amsterdam, 1986.

9. Badr, O., S. D. Probert and P. O'Callaghan. "Rankine Cycles for Steam Power-plants." *Applied Energy (1990).* 36: 191-231.

10. Sinnott, R. K. *Coulson & Richardson's Chemical Engineering* Vol. 6, 3rd Ed., *Chemical Engineering Design*. Butterworth Heinemann: Oxford.

11. Munson, Bruce R., Donald F. Young, and Theodore H. Okiishi. *Fundamentals of Fluid Mechanics*, 3rd Ed. John Wiley & Sons: New York, 1998.

12. http://www.usvacuumpumps.com/pump_sizing.html

13. Krane, R. J. Personal Communication. Spring 2001.

14. Graham, Frank. *Power Plant Engineers Guide*, 2nd Ed. Howard W. Sams & Co.: Indianapolis, 1974.

15. Moran, Michael J. and Howard N. Shapiro. *Fundamentals of Engineering Thermodynamics*, 3rd Ed. John Wiley & Sons, Inc.: New York, 1996.

16. *Marks' Standard Handbook for Mechanical Engineers,* 8th edition. Ed. Avallone, Baumeister, Baumeister III. McGraw-Hill: New York, 1978.

17. Azbel, David. *Heat Transfer Applications in Process Engineering*. Noyes Publications: Park Ridge, NJ, 1984.

18. http://www.cheresources.com/exprules.shtml

19. American Society of Heating and Air Conditioning Engineers, Inc. *1997 ASHRAE Handbook: Fundamentals*. Atlanta, GA, 1997.

20. Tabor, H. "Solar Ponds." *Solar Energy*. Vol. 27, No. 3, pp. 181-194.

21. Lu, Huanmin. Electronic Mail Correspondence. March 19, 2001.

22. Burmeister, Louis C. *Elements of Thermal Fluid System Design*. Prentice Hall: New Jersey, 1998.

23. Popper, Herbert. *Modern Cost-Engineering Techniques*. McGraw-Hill: New York, 1970.

24. Krane, R. J. Class lecture. University of Tennessee. Fall, 2000

25. Westney Richard E. *The Engineer's Cost Handbook: Tools for Managing Project Costs*. Marcel Dekker, Inc.: New York, 1997.

26. California Energy Commission. *Emerging Renewable Resources Account*, Volume 3 of *Guidebook: Renewable Technology Program*. State of California: California, 1999.

Appendices

1. MATLAB Codes Used in Design Process
2. Vendor Information
4. PONDFEAS Information

Appendix 1: MATLAB codes used in design process

1. Properties of seawater based on correlations given in Khan (8)
2. Properties of pure water based on correlations given in Badr (9)
3. Properties of pure water based on IAPWS correlations
4. Properties of gas mixtures, including psychrometric relationships
5. Design of system components
6. Design of power cycle, and printout of results

1. Properties of seawater based on correlations given in Khan (8)

Several codes were developed to analyze the properties of seawater based upon the correlations given by Khan (8).

Subroutines include:
1. sbartos.m -- convert salinity from ppm to gm/kg
2. stosbar.m -- convert salinity from gm/kg to ppm
3. seacp.m -- calculate the specific heat capacity of brine
4. seadensity.m -- calculate the density of brine
5. seadynvisc.m -- calculate the dynamic viscosity of brine
6. seaenthalpy.m -- calculate the enthalpy of brine
7. vaporpressure.m -- calculate the vapor pressure of brine and fresh water

```
function [s]=sbartos(sbar)
% This code converts the salinity of salt between the two
% representations.
% sbar--parts per million
% s--gm/kg
Ms=75.42;
Mw=18.016;
s = 10^3/(1+((10^6/sbar)-1)*(Mw/Ms));
```

```
function [sbar]=stosbar(s)
% This code converts the salinity of salt between the two
% representations.
% sbar--parts per million
% s--gm/kg
Ms=75.42;
Mw=18.016;
sbar = 10^6/(1+(Ms/Mw)*(10^3/s-1));
```

```
function [cp]=seacp(S,t)
% Calculate the specific heat capacity of sea water.
% Valid for 0 to 160 gm/kg and 0 to 180 deg. C
% S in gm/kg
% t in deg. C
% cp in J/kg.K
A=4206.8-6.6197*S+1.2288e-2*(S^2);
B=-1.1262+5.4178e-2*S-2.2719e-4*(S^2);
C=1.2026e-2-5.3566e-4*S+1.8906e-6*(S^2);
D=6.8774e-7+1.517e-6*S-4.4268e-9*S^2;
cp = A + B*t + C*t^2 + d*t^3;
```

```
function [d]=seadensity(S,t)
% This function calculates the density of seawater based on
% salinity and temperature.
% Valid for 0 to 160 gm/kg and 10 to 180 deg. C
% S--gm/kg
```

```
% t--deg. C
% density is in gm/cm^3
X=(2*S-150)/150;
a0 = 2.016110 + 0.115313*X + (0.000326)*(2*X^2 - 1);
a1 = -0.0541  + 0.001571*X - (0.000423)*(2*X^2 - 1);
a2 = -0.006124+ 0.001740*X - (0.000009)*(2*X^2 - 1);
a3 = 0.000346 + 0.000087*X - (0.000053)*(2*X^2 - 1);
Y = (2*t-200)/160;
d = 0.5*a0 + a1*Y + a2*(2*Y^2 -1) + a3*(4*Y^3 -3*Y);
```

```
function [v]=seadynvisc(S,t)
% Dynamic viscosity of water in centipoise
%  % Valid for 0 to 130 gm/kg and 10 to 150 deg. C
% note:  centipoise = 3.6 kg/m.h
% t in deg C
% S in gm/kg
nw = exp(-3.79418 + 604.129/(139.18+t));
a1 = 1.474e-3 + 1.5e-5*t - 3.927e-8*t^2;
a2 = 1.0734e-5 - 8.5e-8*t + 2.23e-10*t^2;
nr = 1+a1*S +a2*S^2;
v = nw*nr;  % use this result for centipoise
% v = v*3.6; % use this result for kg/m.h
```

```
function [h]=seaenthalpy(S,T)
% Calculate the specific enthalpy of seawater.
% Valid for 0 to 160 gm/kg and 0 to 180 deg. C
% S in gm/kg
% T in deg. C
% h in  J/kg
A=4206.8-6.6197*S+1.2288e-2*(S^2);
B=-1.1262+5.4178e-2*S-2.2719e-4*(S^2);
C=1.2026e-2-5.3566e-4*S+1.8906e-6*(S^2);
D=6.8774e-7+1.517e-6*S-4.4268e-9*S^2;
h0=2.3e-3*S-1.03e-4*S^2;
h=h0+2.38846e-4*(A*T+.5*B*T^2+(C/3)*T^3+.25*D*T^4);
% Convert from kcal/kg to J/kg, 1 cal=4.1868 J
h=h*4186.8;
```

```
function [pw,p] = vaporpressure(S,t)
% Calculate the vapor pressure of pure water and sea water
% Valid for 0 to 160 gm/kg and 0 to 200 deg. C
% pw is vapor pressure of pure water, bar
% p is vapor pressure of seawater, bar
% S is salinity, gm/kg
% t is temperature, deg. C
T = t+273.15;
pk = 220.93;
Tk = 647.25;

b(1) = -7.8889166;
```

```
b(2) = 2.5514255;
b(3) = -6.7161690;
b(4) = 33.239495;
b(5) = -105.38479;
b(6) = 174.35319;
b(7) = -148.39348;
b(8) = 48.631602;

sum = 0;
for i = 1:8
   temp1 = (1-T/Tk);
   temp2 = .5*(i+1);
   sum = sum + (b(i))*(temp1^temp2);
end
sum = (Tk/T)*sum;
pw = pk*exp(sum);

p = pw*10^(-2.1609e-4*S - 3.5012e-7*S^2);
```

2. Properties of pure water based on correlations given in Badr (9)

The following subroutines are based on the properties data found in:

Badr, O., S. D. Probert and P. O'Callaghan. "Rankine Cycles for Steam Power-plants." *Applied Energy (1990).* 36: 191-231.

The subroutines include:

1. compressed.m – properties of compressed water
2. saturatedwater.m – properties of saturated water
3. saturatedsteam.m – properties of saturated steam
4. superheat.m – properties of superheated steam
5. quality.m – properties of water/steam mixture under vapor dome

```
function [V,H,S]=compressed(T,CP);
%  Let's calculate the thermodynamic properties of water and steam.
%  The equations are straight from the Badr paper that was handed out
%  in class.
%
%  This code is for thermodynamic properties of compressed water at
%  the input conditions:
%  T, deg. C
%  CP, bar
%
%  The output of the code has the following units:
%  V, m^3/kg
%  H, J/kg
%  S, J/kg.K
%

if CP > 221.1
   disp('Error.  Beyond supercritical pressure.');
   return;
end

T = T+273.15;
CP = CP * 10^5;
P = exp(23.196452-3816.44/(T-46.13));
if CP < P
   disp('Error.  Out of range of interest.');
   disp('    Saturation pressure (Pa) at given temperature is ');
   P=P
   return;
end
if T > 647.3
   disp('Error.  Out of range of interest.');
   disp('     Critical temperature is 374.15 deg. C');
   return;
end

PS(1) = exp(23.196452-3816.44/(T-46.13));
if T > 442
   for i = 2 : 20
```

```
      j = i-1;
      F1 = log(PS(j))- 60.228852+6869.5/T+5.115*log(T)-(7.875e-
3)*PS(j)/(T^2);
      F2 = 1./PS(j)-(7.875e-3)/T^2;
      PS(i) = PS(j)-F1/F2;
      del = abs(PS(i)-PS(j));
      if del < .1 | i == 20
         break;
      end
   end
   P1 = PS(i);
else
   P1 = PS(1);
end
T1 = T;

B1 = (2641.62*10^(80870./(T1^2)))/T1;
B0 = 1.89 - B1;
B2 = 82.546;
B3 = 162460./T1;
B4 = 0.21828*T1;
B5 = 126970./T1;
F0 = 1.89 - B1*(2. + 372740./(T1^2));
B6 = B0 * B3 - 2.* F0 * (B2-B3);
B7 = 2. * F0 * (B4-B5)- B0*B5;
F = 1804036.3+1472.265*T1 + .37789824*(T1^2)+47845.137*log(T1);
BE = (B0-F0)*P1/101325;
BA = B0*(P1/(101325.*T1))^2/2.;
BT = B6+(B0*P1/(101325*T1))^2*(B0*(B4-B5)-2.*B7)/2.;
BB = (BE + BA * BT)/T1;
B = B0*(1.+B0*P1*(B2-B3+((B0*P1/(101325.*T1))^2)*(B4-
B5))/(101325.*(T1^2)));
H1 = (P1/(101325.*T1))^2*(B0*(B2-B3+B0*B7*((P1/(101325.*T1))^2))-B6);
H1 = F + 101.31358*(F0*P1/101325. + H1*B0/2.);
S1 = 1472.2626*log(T1)-461.4874*log(P1)+0.7557174*T1;
S1 = S1+3830.4065-47845.076/T1-101.31344*BB;

G = 0.29607-(1.3973428e-4)*T-(1.1556161e-7)*(T^2);
if T > 518.
   VR = 1.+0.5645828*sqrt(1.-T/647.3)*log(1.-T/647.3);
   VR = VR - 0.50879*(1.-T/647.3)-0.91534*((1.-T/647.3)^2);
else
   VR = 0.33593-(5.2453267e-4)*T+(3.6263003e-6)*(T^2);
   VR = VR - (7.4667901e-9)*(T^3) + (6.346708e-12)*(T^4);
end
VF = (3.104304e-3)*VR*(1.-0.334*G);
if T < 450
   if T < 374
      DH = H1 - 4186.8*(T-273.15);
   else
      XI = 1.3615467*(643.3-T)/T;
      DH = 6051.1583*T*(XI+XI^(0.35298))/(1.+XI^(0.13856));
   end
else
   DH = 2115173.3*(1.-T/647.3)^(.354)+1125343.9*(1.-T/647.3)^(0.456);
end
HF = H1 - DH;
```

```
C1 = 6.9547-0.1178515*T+(4.5658052e-4)*T^2;
C1 = C1 - (7.5049588e-7)*T^3 + (4.7142686e-10)*T^4;
CN = 0.69384*exp(C1);
VL = VF/((1. + (9.347651e-8)*CN*(CP - P))^(0.1111111));
CT = 46.13+3816.44/(23.196452-log(CP));
T1 = CT;
P1 = CP;

B1 = (2641.62*10^(80870./(T1^2)))/T1;
B0 = 1.89 - B1;
B2 = 82.546;
B3 = 162460./T1;
B4 = 0.21828*T1;
B5 = 126970./T1;
F0 = 1.89 - B1*(2. + 372740./(T1^2));
B6 = B0 * B3 - 2.* F0 * (B2-B3);
B7 = 2. * F0 * (B4-B5)- B0*B5;
F = 1804036.3+1472.265*T1 + .37789824*(T1^2)+47845.137*log(T1);
BE = (B0-F0)*P1/101325;
BA = B0*(P1/(101325.*T1))^2/2.;
BT = B6+(B0*P1/(101325*T1))^2*(B0*(B4-B5)-2.*B7)/2.;
BB = (BE + BA * BT)/T1;
B = B0*(1.+B0*P1*(B2-B3+((B0*P1/(101325.*T1))^2)*(B4-
B5))/(101325.*(T1^2)));
H1 = (P1/(101325.*T1))^2*(B0*(B2-B3+B0*B7*((P1/(101325.*T1))^2))-B6);
H1 = F + 101.31358*(F0*P1/101325. + H1*B0/2.);
S1 = 1472.2626*log(T1)-461.4874*log(P1)+0.7557174*T1;
S1 = S1+3830.4065-47845.076/T1-101.31344*BB;

if CT < 450
   if CT < 374
      CD = H1 - 4186.8*(CT-273.15);
   else
      X = 1.3615467*(647.3-CT)/CT;
      CD = 6051.1583*CT*(X+X^(0.35298))/(1.+X^(0.13856));
   end
else
   CD = 2115173.3*(1.-CT/647.3)^(.354)+1125343.9*(1.-CT/647.3)^(0.456);
end
SL = CD/CT;
SL = S1-SL;
SL = SL+4186.8*log(T/CT);
V = VL;
H = HF;
S = SL;
```

```
function [V,H,S,P]=saturatedwater(T)
%  Let's calculate the thermodynamic properties of water and steam.
%  The equations are straight from the Badr paper that was handed out
%  in class.
%
%  This code is for thermodynamic properties of saturated water at
%  T (deg. C)
%
%  The output of the code has the units:
%  P, Pa
%  V, m^3/kg
%  H, J/kg
```

```
%  S, J/kg.K
%
T = 273.15 + T;
if T > 647.3
   disp('Out of saturation region');
   disp('Critical Temperature is 374.15 deg C');
   return;
end

PS(1) = exp(23.196452-3816.44/(T-46.13));
if T > 442
   for i = 2 : 20
      j = i-1;
      F1 = log(PS(j))- 60.228852+6869.5/T+5.115*log(T)-(7.875e-
3)*PS(j)/(T^2);
      F2 = 1./PS(j)-(7.875e-3)/T^2;
      PS(i) = PS(j)-F1/F2;
      del = abs(PS(i)-PS(j));
      if del < .1 | i == 20
         break;
      end
   end
   P1 = PS(i);
else
   P1 = PS(1);
end
P = P1;

T1 = T;
P1 = P;

B1 = (2641.62*10^(80870./(T1^2)))/T1;
B0 = 1.89 - B1;
B2 = 82.546;
B3 = 162460./T1;
B4 = 0.21828*T1;
B5 = 126970./T1;
F0 = 1.89 - B1*(2. + 372740./(T1^2));
B6 = B0 * B3 - 2.* F0 * (B2-B3);
B7 = 2. * F0 * (B4-B5)- B0*B5;
F = 1804036.3+1472.265*T1 + .37789824*(T1^2)+47845.137*log(T1);
BE = (B0-F0)*P1/101325;
BA = B0*(P1/(101325.*T1))^2/2.;
BT = B6+(B0*P1/(101325*T1))^2*(B0*(B4-B5)-2.*B7)/2.;
BB = (BE + BA * BT)/T1;
B = B0*(1.+B0*P1*(B2-B3+((B0*P1/(101325.*T1))^2)*(B4-
B5))/(101325.*(T1^2)));
H1 = (P1/(101325.*T1))^2*(B0*(B2-B3+B0*B7*((P1/(101325.*T1))^2))-B6);
H1 = F + 101.31358*(F0*P1/101325. + H1*B0/2.);
S1 = 1472.2626*log(T1)-461.4874*log(P1)+0.7557174*T1;
S1 = S1+3830.4065-47845.076/T1-101.31344*BB;

G = 0.29607-(1.3973428e-4)*T-(1.1556161e-7)*(T^2);
if T > 518.
   VR = 1.+0.5645828*sqrt(1.-T/647.3)*log(1.-T/647.3);
   VR = VR - 0.50879*(1.-T/647.3)-0.91534*((1.-T/647.3)^2);
else
```

```
   VR = 0.33593-(5.2453267e-4)*T+(3.6263003e-6)*(T^2);
   VR = VR - (7.4667901e-9)*(T^3) + (6.346708e-12)*(T^4);
end
VF = (3.104304e-3)*VR*(1.-0.334*G);
if T < 450
   if T < 374
      DH = H1 - 4186.8*(T-273.15);
   else
      XI = 1.3615467*(643.3-T)/T;
      DH = 6051.1583*T*(XI+XI^(0.35298))/(1.+XI^(0.13856));
   end
else
   DH = 2115173.3*(1.-T/647.3)^(.354)+1125343.9*(1.-T/647.3)^(0.456);
end
HF = H1 - DH;

DS = DH/T;
SF = S1 - DS;

V = VF;
H = HF;
S = SF;
```

```
function [V,H,S,P]=saturatedsteam(T)
%  Let's calculate the thermodynamic properties of water and steam.
%  The equations are straight from the Badr paper that was handed out
%  in class.
%
%  This code is for thermodynamic properties of saturated steam at
%  T (deg. C)
%
%  The output of the code has the following units:
%  P, Pa
%  V, m^3/kg
%  H, J/kg
%  S, J/kg.K
%
T = 273.15 + T;
if T > 647.3
   disp('Out of saturation region');
   disp('Critical Temperature is 374.15 deg C');
   return;
end

PS(1) = exp(23.196452-3816.44/(T-46.13)); %  Pressure in Pa
if T > 442
     for i = 2 : 20
      j = i-1;
      F1 = log(PS(j)) - 60.228852 + 6869.5/T + 5.115*log(T)...
         - (7.875e-3)*PS(j)/(T^2);
      F2 = (1./PS(j))-((7.875e-3)/(T^2));
      PS(i) = PS(j) - F1/F2;
      del = abs(PS(i)-PS(j));
      if del < .1 | i == 20
         break;
      end
   end
   P1 = PS(i);
```

```
else
   P1 = PS(1);
end
P = P1;

T1 = T;
P1 = P;

B1 = (2641.62*10^(80870./(T1^2)))/T1;
B0 = 1.89 - B1;
B2 = 82.546;
B3 = 162460./T1;
B4 = 0.21828*T1;
B5 = 126970./T1;
F0 = 1.89 - B1*(2. + 372740./(T1^2));
B6 = B0 * B3 - 2.* F0 * (B2-B3);
B7 = 2. * F0 * (B4-B5)- B0*B5;
F = 1804036.3+1472.265*T1 + .37789824*(T1^2)+47845.137*log(T1);
BE = (B0-F0)*P1/101325;
BA = B0*(P1/(101325.*T1))^2/2.;
BT = B6+(B0*P1/(101325*T1))^2*(B0*(B4-B5)-2.*B7)/2.;
BB = (BE + BA * BT)/T1;
B = B0*(1.+B0*P1*(B2-B3+((B0*P1/(101325.*T1))^2)*(B4-
B5))/(101325.*(T1^2)));
H1 = (P1/(101325.*T1))^2*(B0*(B2-B3+B0*B7*((P1/(101325.*T1))^2))-B6);
H1 = F + 101.31358*(F0*P1/101325. + H1*B0/2.);
S1 = 1472.2626*log(T1)-461.4874*log(P1)+0.7557174*T1;
S1 = S1+3830.4065-47845.076/T1-101.31344*BB;

VG = (1.000035e-3)*(461539.43*T1/P1+B);
V=VG;
H = H1;
S = S1;
```

```
function [V,H,S]=superheat(T,P)
%  Let's calculate the thermodynamic properties of water and steam.
%  The equations are straight from the Badr paper that was handed out
%  in class.
%
%  This code is for thermodynamic properties of superheated steam at
%  T (deg. C) and CP (bar)
%
%  The output of the code has the following units:
%  V, m^3/kg
%  H, J/kg
%  S, J/kg.K
%

if P>221.2
   disp('This is supercritical pressure and beyond this code.');
   return;
end
T2 = T + 273.15;
P2 = P*1e5;
P3 = exp(23.196452-3816.44/(T2-46.13));

if P2 > P3
   disp('Out of superheated range!');
```

```
   disp('Saturation pressure:');
   P=P3/1e5
   return;
end
T1 = T2;
P1 = P2;

B1 = (2641.62*10^(80870./(T1^2)))/T1;
B0 = 1.89 - B1;
B2 = 82.546;
B3 = 162460./T1;
B4 = 0.21828*T1;
B5 = 126970./T1;
F0 = 1.89 - B1*(2. + 372740./(T1^2));
B6 = B0 * B3 - 2.* F0 * (B2-B3);
B7 = 2. * F0 * (B4-B5)- B0*B5;
F = 1804036.3+1472.265*T1 + .37789824*(T1^2)+47845.137*log(T1);
BE = (B0-F0)*P1/101325;
BA = B0*(P1/(101325.*T1))^2/2.;
BT = B6+(B0*P1/(101325*T1))^2*(B0*(B4-B5)-2.*B7)/2.;
BB = (BE + BA * BT)/T1;
B = B0*(1.+B0*P1*(B2-B3+((B0*P1/(101325.*T1))^2)*(B4-
B5))/(101325.*(T1^2)));
H1 = (P1/(101325.*T1))^2*(B0*(B2-B3+B0*B7*((P1/(101325.*T1))^2))-B6);
H1 = F + 101.31358*(F0*P1/101325. + H1*B0/2.);
S1 = 1472.2626*log(T1)-461.4874*log(P1)+0.7557174*T1;
S1 = S1+3830.4065-47845.076/T1-101.31344*BB;

VG = (1.000035e-3)*(461539.43*T1/P1+B);
V=VG;
H = H1;
S = S1;
```

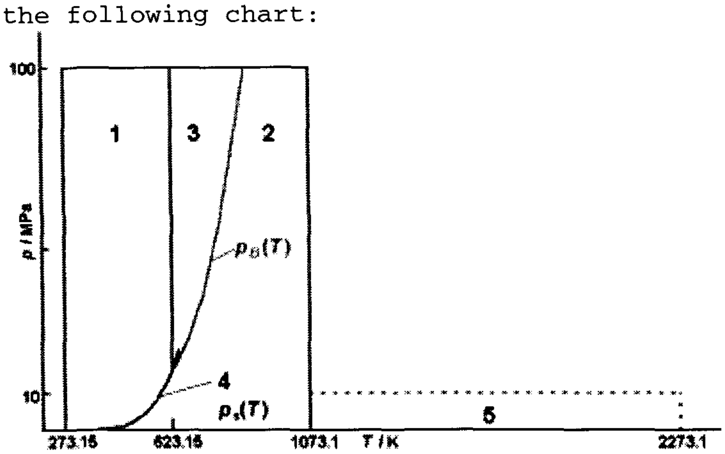
```
function [V,H,S,P]=quality(T,x)
%  This code evaluates V, H, and S under the liquid/vapor dome
%  by averaging the values.
%
%  Input values:
%  T -- water temperature, deg. C
%  x -- steam quality, (1.0 = all steam, 0.0 = all liquid)
%
%  Output values:
%  V -- specific volume, m^3/kg
%  H -- specific enthalpy, kJ/kg
%  S -- specific entropy, kJ/kg.K
%  P -- saturation pressure, Pa

[VW,HW,SW,P]=saturatedwater(T);
[VS,HS,SS,P]=saturatedsteam(T);

V = VW +x*(VS - VW);
H = HW +x*(HS - HW);
S = SW +x*(SS - SW);
```

3. Properties of pure water based on IAPWS correlations

The following subroutines are based on the compilation of IAPWS-IF97 property data found on .pdf files published at the Chemical Engineering section of about.com. Specific web pages referenced include:

Introduction:
http://chemengineer.about.com/science/chemengineer/library/weekly/aa071700a.htm
Thermodynamic properties:
http://chemengineer.about.com/science/chemengineer/library/weekly/aa073100a.htm
Transport properties:
http://chemengineer.about.com/science/chemengineer/library/weekly/aa081400a.htm

The different regions mentioned in the codes can be visualized using the following chart:



Subroutines included in this section are:

1. whichregion.m - Routine which calls region1.m, region2.m, or region3.m as appropriate based on the region temperature and pressure boundaries
2. region1.m - Thermodynamic properties for region 1
3. gamma1.m - Coefficients for region 1
4. region2.m - Thermodynamic properties for region 2
5. gamma02.m - Coefficients for region 2
6. gammar2.m - Coefficients for region 2
7. region3.m - Thermodynamic properties of water in region 3
8. phi.m - Coefficients for region 3
9. mu.m - Viscosity of water at a given temperature and pressure
10. surftens.m - Surface tension of water at a given temperature
11. watersattemp.m - Water saturation temperature as a function of saturation pressure
12. watersatpres.m - Water saturation pressure as a function of temperature

```
function [r,v,u,h,s,cp,cv,eta,kinvis,k] = whichregion(T,p)
% This code combines the codes region1.m, region2.m, and region3.m
% to give property values regardless of the input conditions.
%
```

```
%  Caution: No values are assigned in the saturation region.
%
%  Input parameters are given by:
%     T -- Temperature, K
%     p -- Pressure, MPa
%  Output parameters:
%     r -- The IAPWS region
%     v -- specific volume, m^3/kg
%     u -- specific internal energy, kJ/kg
%     s -- specific entropy, kJ/kg.K
%     h -- specific enthalpy, h = u + pv, kJ/kg
%     cp -- specific heat at constant pressure, kJ/kg.K
%     cv -- specific heat at constant volume, kJ/kg.K
%     eta -- dynamic viscosity, Pa.s
%     kinvis -- kinematic viscosity, m^2/s
%     k -- thermal conductivity, W/m.K

%  Coefficients of boundary equation for p_B
nb = [0.34805185628969E+03
-0.11671859879975E+01
0.10192970039326E-02
0.57254459862746E+03
0.13918839778870E+02];
%  Boundary Equations
p_B = nb(1) + nb(2)*T + nb(3)*T^2;
T_B = nb(4) + sqrt((p-nb(5))/nb(3));

%  Coefficients of boundary equation for saturation curve
%  273.15 K < T < 647.096 K or 611.213 Pa < p < 22.064 MPa
ns = [0.11670521452767E+04
-0.72421316703206E+06
-0.17073846940092E+02
0.12020824702470E+05
-0.32325550322333E+07
0.14915108613530E+02
-0.48232657361591E+04
0.40511340542057E+06
-0.23855557567849E+00
0.65017534844798E+03];
%  Vapor-Liquid Saturation Curve equations
nu = T + ns(9)/(T - ns(10));
A = nu^2 + ns(1)*nu + ns(2);
B = ns(3)*nu^2 + ns(4)*nu + ns(5);
C = ns(6)*nu^2 + ns(7)*nu + ns(8);
ps = ((2*C)/((-B+sqrt(B^2-4*A*C))))^4;
beta = p^.25;
E = beta^2 + beta*ns(3) + ns(6);
F = (beta^2)*ns(1) + beta*ns(4) + ns(7);
G = (beta^2)*ns(2) + beta*ns(5) + ns(8);
D = (2*G)/(-1*F - sqrt(F^2 - 4*E*G));
Ts = .5*( ns(10) + D - sqrt( (ns(10)+D)^2 - 4*(ns(9) + ns(10)*D) ) );

if T == Ts & p == ps
   r = 4;
   disp('saturated region');
elseif (T > 273.15 & T < 623.15)
   if (p > ps & p < 100)
```

```
      r = 1;
      [gibbs,v,u,h,s,cp,cv,eta,kinvis,k] = region1(T,p);
   elseif (p > 0 & p < ps)
      r = 2;
      [v,u,s,h,cp,cv,g,eta,kinvis,k] = region2(T,p);
   end
elseif (T > 623.15 & T < 863.15) & (p > 0 & p < ps)
   r = 2;
   [v,u,s,h,cp,cv,g,eta,kinvis,k] = region2(T,p);
elseif (T > 863.15 & T < 1073.15) & (p > 0 & p < 100)
   r = 2;
   [v,u,s,h,cp,cv,g,eta,kinvis,k] = region2(T,p);
elseif (T > 623.15 & T < T_B) & (p > p_B & p < 100)
   r = 3;
   [v,u,s,h,cp,cv,f,eta,kinvis,k] = region3(T,p);
else
   disp('error out of range');
end
```

```
function [gibbs,v,u,h,s,cp,cv,eta,kinvis,k] = region1(T,p)
%   Region 1 of the IAPWS-IF97 is 273.15K<T<623.16 and ps(T) < p <
100Mpa
%
%  Input parameters are given by:
%     T in K
%     p in MPa
%  Output parameters:
%     v -- specific volume, m^3/kg
%     u -- specific internal energy, kJ/kg
%     s -- specific entropy, kJ/kg.K
%     h -- specific enthalpy, h = u + pv, kJ/kg
%     cp -- specific heat at constant pressure, kJ/kg.K
%     cv -- specific heat at constant volume, kJ/kg.K
%     g -- specific Gibb's function, kJ/kg
%     eta -- dynamic viscosity, Pa.s
%     kinvis -- kinematic viscosity, m^2/s
%     k -- thermal conductivity, W/m.K

[gamma,gammap,gammapp,gammat,gammatt,gammapt] = gamma1(T,p);
R = 0.461526;  %kJ/kg.K
tau = 1386/T;
pr = p/16.53;

gibbs = R * T * gamma;  % kJ/kg
v = (pr * gammap * R * T)/(p*1000); % m^3/kg
u = (tau*gammat - pr*gammap)*R*T;   % kJ/kg
h = u + (p*1000)*v;                 % kJ/kg, h = u + pv
s = (tau*gammat - gamma)*R ;        % kJ/kg.K
cp = (-1*(tau^2)*gammatt)*R;        % kJ/kg.K
cv = (-1*(tau^2)*gammatt + (1/gammapp)*(gammap - tau*gammapt)^2)*R;

%  Dimensionless variables for calculating dynamic viscosity, eta
etastar = 55.071e-6; %  Pa.s
rho = 1/v;
del = rho/317.763;
tau2 = 647.226/T;

%  Ideal gas part of dynamic viscosity
```

```
n0 = [.1e1 .978197 .579829 -.202354];
Y0 = 0;
for i = 1:4
   Y0 = Y0 + (n0(i))*tau2^(i-1);
end
Y0 = ((tau2^.5)*(Y0))^-1;

%  Real fluid part of dynamic viscosity
temp = [1 0 0 0.5132047
        2 0 1 0.3205656
        3 0 4 -0.7782567
        4 0 5 0.1885447
        5 1 0 0.2151778
        6 1 1 0.7317883
        7 1 2 0.1241044E+01
        8 1 3 0.1476783E+01
        9 2 0 -0.2818107
       10 2 1 -0.1070786E+01
       11 2 2 -0.1263184E+01
       12 3 0 0.1778064
       13 3 1 0.4605040
       14 3 2 0.2340379
       15 3 3 -0.4924179
       16 4 0 -0.4176610E-01
       17 4 3 0.1600435
       18 5 1 -0.1578386E-01
       19 6 3 -0.3629481E-02];
Id = temp(:,2);
Jd = temp(:,3);
nd = temp(:,4);
Y1 = 0;
for i = 1:19
   Y1 = Y1 + nd(i) * ((del - 1)^Id(i)) * ((tau2 - 1)^Jd(i));
end
Y1 = exp(del * Y1);
Y = Y0*Y1;
eta = etastar*Y;
kinvis = eta * v;

%  Coefficients of the ideal gas part for thermal conductivity
C0 = [0.1000000E+01
0.6978267E+01
0.2599096E+01
-0.9982540];
%  Coefficients of the first real fluid part for thermal
%     conductivity (i's across and j's down)
C1 = [0.13293046E+01  0.17018363E+01  0.52246158E+01  0.87127675E+01 -
0.18525999E+01
     -0.40452437     -0.22156845E+01 -0.10124111E+02 -0.95000611E+01
0.93404690
      0.24409490      0.16511057E+01  0.49874687E+01  0.43786606E+01
0.
      0.18660751E-01 -0.76736002     -0.27297694     -0.91783782
0.
     -0.12961068      0.37283344     -0.43083393      0.
0.
```

```
     0.44809953E-01 -0.11203160      0.13333849      0.
0.];
partial1 = ((647.226*16.53/22.115)*(gammapt*1386 -
gammap*T))/(T^2*gammapp);
partial2 = -1*(22.115*1000*gammapp)/(317.763*R*T*gammap^2);
  V0 = 0;
  for i = 1:4
     V0 = V0 + C0(i)*(tau2^(i-1));
  end
  V0 = (sqrt(tau2)*V0)^-1;

  V1 = 0;
  for i = 1:5
     for j = 1:6
        V1 = V1 + C1(j,i)*(tau2 - 1)^(i-1) * (del - 1)^(j-1);
     end
  end
  V1 = exp(del*V1);

  V2 = (0.0013848/Y)*((tau2*del)^-
2)*(partial1^2)*((del*partial2)^0.4678)*sqrt(del) ...
     *exp(-18.66*((tau2^-1) - 1)^2 - (del-1)^4);
  V = V0*V1 + V2;
  k = V*0.4945;   %  W/m.K
```

```
function [gamma,gammap,gammapp,gammat,gammatt,gammapt] = gamma1(T,p)
%   Region 1 of the IAPWS-IF97 is 273.15K<T<623.16 and ps(T) < p <
100Mpa
%
%  This code gives coefficients for the calculation of thermodynamic
%  properties in region one of IAPWS-IF97 steam tables.
%
%  Input variables:
%  T in Kelvin
%  p in MPa


I = [ 0   0   0   0   0   0   0   0   1   1   1   1   1   1   2   2   2
...
   2   2   3   3   3   4   4   4   5   8   8   21   23   29   30   31
32];
J = [ -2   -1   0   1   2   3   4   5   -9   -7   -1   0   1   3   -3
0 ...
   1   3   17   -4   0   6   -5   -2   10   -8   -11   -6   -29   -31
-38 ...
   -39   -40   -41];
n = [0.14632971213167e0
   -0.84541887169114e0
   -0.37563603672040e1
   0.33855169168385e1
   -0.95791963387872e0
   0.15772038513228e0
   -0.16616417199501e-1
   0.81214629983568e-3
   0.28319080123804e-3
   -0.60706301565874e-3
   -0.18990068218419e-1
   -0.32529748770505e-1
```

```
   -0.21841717175414e-1
   -0.52838357969930e-4
   -0.47184321073267e-3
   -0.30001780793026e-3
   0.47661393906987e-4
   -0.44141845330846e-5
   -0.72694996297594e-15
   -0.31679644845054e-4
   -0.28270797985312e-5
   -0.85205128120103e-9
   -0.22425281908000e-5
   -0.65171222895601e-6
   -0.14341729937924e-12
   -0.40516996860117e-6
   -0.12734301741641e-8
   -0.17424871230634e-9
   -0.68762195531e-18
   0.14478307828521e-19
   0.26335781662795e-22
   -0.11947622640071e-22
   0.18228094581404e-23
   -0.93537087292458e-25];

tau = 1386/T;
pr = p/16.53;

gamma = 0;
for i = 1:34
   gamma = gamma + n(i)*((7.1-pr)^I(i))*((tau - 1.222)^J(i));
end

gammap = 0;
for i = 1:34
   gammap = gammap + -1*n(i)*I(i)*((7.1-pr)^(I(i)-1))*((tau -
1.222)^J(i));
end

gammapp = 0;
for i = 1:34
   gammapp = gammapp + n(i)*I(i)*(I(i)-1)*((7.1-pr)^(I(i)-2))*((tau -
1.222)^J(i));
end

gammat = 0;
for i = 1:34
   gammat = gammat + n(i)*((7.1 - pr)^I(i))*J(i)*((tau-1.222)^(J(i)-
1));
end

gammatt = 0;
for i = 1:34
   gammatt = gammatt + ...
      n(i) * ((7.1 - pr)^(I(i))) * J(i) * (J(i)-1) * ((tau -
1.222)^(J(i)-2));
end

gammapt = 0;
```

```
for i = 1:34
   gammapt = gammapt + -1*n(i)*I(i)*((7.1-pr)^(I(i)-1))*J(i)*((tau -
1.222)^(J(i)-1));
end
```

```
function [v,u,s,h,cp,cv,g,eta,kinvis,k] = region2(T,p)
%  Region two for the IAPWS-IF97 property equations is
%  defined by:
%  273.15 K < T < 623.15 K and 0 < p < ps(T)
%  623.15 K < T < 863.15 K and 0 < p < p_B(T)
%  863.15 K < T < 1073.15 K and 0 < p < 100 MPa
%
%  Input parameters are given by:
%     T in K
%     p in MPa
%  Output parameters:
%     v -- specific volume, m^3/kg
%     u -- specific internal energy, kJ/kg
%     s -- specific entropy, kJ/kg.K
%     h -- specific enthalpy, h = u + pv, kJ/kg
%     cp -- specific heat at constant pressure, kJ/kg.K
%     cv -- specific heat at constant volume, kJ/kg.K
%     g -- specific Gibb's function, kJ/kg
%     eta -- dynamic viscosity, Pa.s
%     kinvis -- kinematic viscosity, m^2/s
%     k -- thermal conductivity, W/m.K

%  Determine coefficients for calculating property values:
[gamma0,gamma0p,gamma0pp,gamma0t,gamma0tt,gamma0pt] = gamma02(T,p);
[gammar,gammarp,gammarpp,gammart,gammartt,gammarpt] = gammar2(T,p);

R = 0.461526;  % kJ/kg.K
tau = 540/T;

v = (R * T * p *(gamma0p + gammarp))/(p*1000);
u = (R * T)*(tau*(gamma0t + gammart) - p*(gamma0p + gammarp));
s = R * (tau*(gamma0t+gammart) - (gamma0 + gammar));
h = R * T *(tau *(gamma0t + gammart));
cp = R*-1*(tau^2)*(gamma0tt + gammartt);
cv = -1*tau^2*(gamma0tt + gammartt) - ...
   ((1 + p*gammarp - tau*p*gammarpt)^2)/(1-p^2*gammarpp);
cv = cv*R;
g = R * T *(gamma0 + gammar);

%  Dimensionless variables for calculating dynamic viscosity, eta
etastar = 55.071e-6; %  Pa.s
rho = 1/v;
del = rho/317.763;
tau2 = 647.226/T;

%  Ideal gas part of dynamic viscosity
n0 = [.1e1 .978197 .579829 -.202354];
Y0 = 0;
for i = 1:4
   Y0 = Y0 + (n0(i))*tau2^(i-1);
end
Y0 = ((tau2^.5)*(Y0))^-1;
```

```
%  Real fluid part of dynamic viscosity
temp = [1 0 0 0.5132047
        2 0 1 0.3205656
        3 0 4 -0.7782567
        4 0 5 0.1885447
        5 1 0 0.2151778
        6 1 1 0.7317883
        7 1 2 0.1241044E+01
        8 1 3 0.1476783E+01
        9 2 0 -0.2818107
       10 2 1 -0.1070786E+01
       11 2 2 -0.1263184E+01
       12 3 0 0.1778064
       13 3 1 0.4605040
       14 3 2 0.2340379
       15 3 3 -0.4924179
       16 4 0 -0.4176610E-01
       17 4 3 0.1600435
       18 5 1 -0.1578386E-01
       19 6 3 -0.3629481E-02];
Id = temp(:,2);
Jd = temp(:,3);
nd = temp(:,4);
Y1 = 0;
for i = 1:19
   Y1 = Y1 + nd(i) * ((del - 1)^Id(i)) * ((tau2 - 1)^Jd(i));
end
Y1 = exp(del * Y1);
Y = Y0*Y1;
eta = etastar*Y;
kinvis = eta * v;

%  Coefficients of the ideal gas part for thermal conductivity
C0 = [0.1000000E+01
0.6978267E+01
0.2599096E+01
-0.9982540];
%  Coefficients of the first real fluid part for thermal
%     conductivity (i's across and j's down)
C1 = [0.13293046E+01  0.17018363E+01  0.52246158E+01  0.87127675E+01 -
0.18525999E+01
     -0.40452437     -0.22156845E+01 -0.10124111E+02 -0.95000611E+01
0.93404690
      0.24409490      0.16511057E+01  0.49874687E+01  0.43786606E+01
0.
      0.18660751E-01 -0.76736002     -0.27297694     -0.91783782
0.
     -0.12961068      0.37283344     -0.43083393      0.
0.
      0.44809953E-01 -0.11203160      0.13333849      0.
0.];
   partial1 = (647.226*1/22.115)*((gamma0pt + gammarpt)*540 - ...
      (gamma0p + gammarp)*T)/((gamma0pp + gammarpp)*T^2);
   partial2 = (-1*(22.115*1000)*(gamma0pp + gammarpp))/...
      (317.763*R*T*(gamma0p+gammarp)^2);
  V0 = 0;
  for i = 1:4
```

```
    V0 = V0 + C0(i)*(tau2^(i-1));
  end
  V0 = (sqrt(tau2)*V0)^-1;

  V1 = 0;
  for i = 1:5
     for j = 1:6
        V1 = V1 + C1(j,i)*(tau2 - 1)^(i-1) * (del - 1)^(j-1);
     end
  end
  V1 = exp(del*V1);

  V2 = (0.0013848/Y)*((tau2*del)^-
2)*(partial1^2)*((del*partial2)^0.4678)*sqrt(del) ...
     *exp(-18.66*((tau2^-1) - 1)^2 - (del-1)^4);
  V = V0*V1 + V2;
  k = V*0.4945;    %  W/m.K
```

```
function [gamma0,gamma0p,gamma0pp,gamma0t,gamma0tt,gamma0pt] =
gamma02(T,p)
%  Region two for the IAPWS-IF97 property equations is
%  defined by:
%  273.15 K < T < 623.15 K and 0 < p < ps(T)
%  623.15 K < T < 863.15 K and 0 < p < p_B(T)
%  863.15 K < T < 1073.15 K and 0 < p < 100 MPa

%  T in K
%  p in MPa

%  Coefficients of ideal gas part of fundamental equation
n0 = [-0.96927686500217E+01
0.10086655968018E+02
-0.56087911283020E-02
0.71452738081455E-01
-0.40710498223928E+00
0.14240819171444E+01
-0.43839511319450E+01
-0.28408632460772E+00
0.21268463753307E-01];

J0 = [0 1 -5 -4 -3 -2 -1 2 3];

tau = 540/T;

gamma0 = log(p);
for i = 1:9
   gamma0 = gamma0 + n0(i)*tau^J0(i);
end

gamma0p = 1/p;

gamma0pp = -1/(p^2);

gamma0t = 0;
for i = 1:9
   gamma0t = gamma0t + n0(i)*J0(i)*tau^(J0(i)-1);
end
```

```
gamma0tt = 0;
for i = 1:9
   gamma0tt = gamma0tt + n0(i)*J0(i)*(J0(i)-1)*tau^(J0(i)-2);
end

gamma0pt = 0;
```

```
function [gammar,gammarp,gammarpp,gammart,gammartt,gammarpt] =
gammar2(T,p)
% Region two for the IAPWS-IF97 property equations is
% defined by:
% 273.15 K < T < 623.15 K and 0 < p < ps(T)
% 623.15 K < T < 863.15 K and 0 < p < p_B(T)
% 863.15 K < T < 1073.15 K and 0 < p < 100 MPa

% Input values:
% T in K
% p in MPa

% Coefficients of the residual part of the fundamental equation and
% its derivatives
temp = [1 0 -0.17731742473213E-02
1 1 -0.17834862292358E-01
1 2 -0.45996013696365E-01
1 3 -0.57581259083432E-01
1 6 -0.50325278727930E-01
2 1 -0.33032641670203E-04
2 2 -0.18948987516315E-03
2 4 -0.39392777243355E-02
2 7 -0.43797295650573E-01
2 36 -0.26674547914087E-04
3 0 0.20481737692309E-07
3 1 0.43870667284435E-06
3 3 -0.32277677238570E-04
3 6 -0.15033924542148E-02
3 35 -0.40668253562649E-01
4 1 -0.78847309559367E-09
4 2 0.12790717852285E-07
4 3 0.48225372718507E-06
5 7 0.22922076337661E-05
6 3 -0.16714766451061E-10
6 16 -0.21171472321355E-02
6 35 -0.23895741934104E+02
7 0 -0.59059564324270E-17
7 11 -0.12621808899101E-05
7 25 -0.38946842435739E-01
8 8 0.11256211360459E-10
8 36 -0.82311340897998E+01
9 13 0.19809712802088E-07
10 4 0.10406965210174E-18
10 10 -0.10234747095929E-12
10 14 -0.10018179379511E-08
16 29 -0.80882908646985E-10
16 50 0.10693031879409E+00
18 57 -0.33662250574171E+00
20 20 0.89185845355421E-24
20 35 0.30629316876232E-12
20 48 -0.42002467698208E-05
```

```
21 21 -0.59056029685639E-25
22 53 0.37826947613457E-05
23 39 -0.12768608934681E-14
24 26 0.73087610595061E-28
24 40 0.55414715350778E-16
24 58 -0.94369707241210E-06];

I = temp(:,1);
J = temp(:,2);
n = temp(:,3);

tau = 540/T;

gammar = 0;
for i = 1:43
   gammar = gammar + n(i)*(p^I(i))*(tau - .5)^J(i);
end

gammarp = 0;
for i = 1:43
   gammarp = gammarp + n(i)*I(i)*(p^(I(i)-1))*(tau -.5)^J(i);
end

gammarpp = 0;
for i = 1:43
   gammarpp = gammarpp + n(i)*I(i)*(I(i)-1)*(p^(I(i)-2))*(tau -
.5)^J(i);
end

gammart = 0;
for i = 1:43
   gammart = gammart + n(i)*(p^I(i))*J(i)*(tau - .5)^(J(i)-1);
end

gammartt = 0;
for i = 1:43
   gammartt = gammartt + n(i)*p^(I(i))*J(i)*(J(i)-1)*(tau-.5)^(J(i)-2);
end

gammarpt = 0;
for i = 1:43
   gammarpt = gammarpt + n(i)*I(i)*(p^(I(i)-1))*J(i)*(tau-.5)^(J(i)-1);
end
```

```
function [v,u,s,h,cp,cv,f,eta,kinvis,k] = region3(T,p)
%  Region three for the IAPWS-IF97 property equations is
%  defined by:
%  623.15 K < T < T_B and p_B(T) < p < 100 MPa
%
%  Input parameters are given by:
%     T in K
%     p in MPa
%  Output parameters:
%     v -- specific volume, m^3/kg
%     u -- specific internal energy, kJ/kg
%     s -- specific entropy, kJ/kg.K
%     h -- specific enthalpy, h = u + pv, kJ/kg
%     cp -- specific heat at constant pressure, kJ/kg.K
```

```
%     cv -- specific heat at constant volume, kJ/kg.K
%     eta -- dynamic viscosity, Pa.s
%     kinvis -- kinematic viscosity, m^2/s
%     k -- thermal conductivity, W/m.K
%     f -- Helmholtz free energy


R = 0.461526;
%  Determine coefficients for calculating fluid properties:
[del,phi,phid,phidd,phit,phitt,phidt] = phi(T,p);
tau = 647.096/T;

v = (del*322)^-1;
u = tau*phit*R*T;
s = (tau * phit - phi)*R;
h = R*T*(tau*phit + del*phid);
cp = R*(-1*tau^2*phitt + (del*phid - ...
   del*tau*phidt)^2/(2*del*phid + phidd*del^2));
cv = -1*R*phitt*tau^2;
f = phi*T*R;    %  helmholtz free energy


%  Dimensionless variables for calculating dynamic viscosity, eta
etastar = 55.071e-6; %  Pa.s
rho = 1/v;
del = rho/317.763;
tau2 = 647.226/T;

%  Ideal gas part of dynamic viscosity
n0 = [.1e1 .978197 .579829 -.202354];
Y0 = 0;
for i = 1:4
   Y0 = Y0 + (n0(i))*tau2^(i-1);
end
Y0 = ((tau2^.5)*(Y0))^-1;

%  Real fluid part of dynamic viscosity
temp = [1 0 0 0.5132047
        2 0 1 0.3205656
        3 0 4 -0.7782567
        4 0 5 0.1885447
        5 1 0 0.2151778
        6 1 1 0.7317883
        7 1 2 0.1241044E+01
        8 1 3 0.1476783E+01
        9 2 0 -0.2818107
       10 2 1 -0.1070786E+01
       11 2 2 -0.1263184E+01
       12 3 0 0.1778064
       13 3 1 0.4605040
       14 3 2 0.2340379
       15 3 3 -0.4924179
       16 4 0 -0.4176610E-01
       17 4 3 0.1600435
       18 5 1 -0.1578386E-01
       19 6 3 -0.3629481E-02];
Id = temp(:,2);
```

```
Jd = temp(:,3);
nd = temp(:,4);
Y1 = 0;
for i = 1:19
   Y1 = Y1 + nd(i) * ((del - 1)^Id(i)) * ((tau2 - 1)^Jd(i));
end
Y1 = exp(del * Y1);
Y = Y0*Y1;
eta = etastar*Y;
kinvis = eta * v;

%  Coefficients of the ideal gas part for thermal conductivity
C0 = [0.1000000E+01
0.6978267E+01
0.2599096E+01
-0.9982540];
%  Coefficients of the first real fluid part for thermal
%     conductivity (i's across and j's down)
C1 = [0.13293046E+01  0.17018363E+01  0.52246158E+01  0.87127675E+01 -
0.18525999E+01
     -0.40452437     -0.22156845E+01 -0.10124111E+02 -0.95000611E+01
0.93404690
      0.24409490      0.16511057E+01  0.49874687E+01  0.43786606E+01
0.
      0.18660751E-01 -0.76736002     -0.27297694     -0.91783782
0.
     -0.12961068      0.37283344     -0.43083393      0.
0.
      0.44809953E-01 -0.11203160      0.13333849      0.
0.];
   partial1 = (647.226/(22.115*1000))*(R/(322*v^2))*(phid-
(647.096/T)*phidt);
   partial2 = (22.115*1000)/(317.763*(1/v)*R*T*(2*phid + ((322*v)^-
1)*phidd));
  V0 = 0;
  for i = 1:4
     V0 = V0 + C0(i)*(tau2^(i-1));
  end
  V0 = (sqrt(tau2)*V0)^-1;

  V1 = 0;
  for i = 1:5
     for j = 1:6
        V1 = V1 + C1(j,i)*(tau2 - 1)^(i-1) * (del - 1)^(j-1);
     end
  end
  V1 = exp(del*V1);

  V2 = (0.0013848/Y)*((tau2*del)^-
2)*(partial1^2)*((del*partial2)^0.4678)*sqrt(del) ...
     *exp(-18.66*((tau2^-1) - 1)^2 - (del-1)^4);
  V = V0*V1 + V2;
  k = V*0.4945;   %  W/m.K
```

```
function [del,phi,phid,phidd,phit,phitt,phidt] = phi(T,p)
%  Region three for the IAPWS-IF97 property equations is
%  defined by:
%  623.15 K < T < T_B and p_B(T) < p < 100 MPa
```

```
% Input values:
% T in K
% p in MPa

R = 0.461526;

% Coefficients for fundamental equation
temp = [1 0 0 0.10658070028513E+01
2 0 0 -0.15732845290239E+02
3 0 1 0.20944396974307E+02
4 0 2 -0.76867707878716E+01
5 0 7 0.26185947787954E+01
6 0 10 -0.28080781148620E+01
7 0 12 0.12053369696517E+01
8 0 23 -0.84566812812502E-02
9 1 2 -0.12654315477714E+01
10 1 6 -0.11524407806681E+01
11 1 15 0.88521043984318E+00
12 1 17 -0.64207765181607E+00
13 2 0 0.38493460186671E+00
14 2 2 -0.85214708824206E+00
15 2 6 0.48972281541877E+01
16 2 7 -0.30502617256965E+01
17 2 22 0.39420536879154E-01
18 2 26 0.12558408424308E+00
19 3 0 -0.27999329698710E+00
20 3 2 0.13899799569460E+01
21 3 4 -0.20189915023570E+01
22 3 16 -0.82147637173963E-02
23 3 26 -0.47596035734923E+00
24 4 0 0.43984074473500E-01
25 4 2 -0.44476435428739E+00
26 4 4 0.90572070719733E+00
27 4 26 0.70522450087967E+00
28 5 1 0.10770512626332E+00
29 5 3 -0.32913623258954E+00
30 5 26 -0.50871062041158E+00
31 6 0 -0.22175400873096E-01
32 6 2 0.94260751665092E-01
33 6 26 0.16436278447961E+00
34 7 2 -0.13503372241348E-01
35 8 26 -0.14834345352472E-01
36 9 2 0.57922953628084E-03
37 9 26 0.32308904703711E-02
38 10 0 0.80964802996215E-04
39 10 1 -0.16557679795037E-03
40 11 26 -0.44923899061815E-04];

I = temp(:,2);
J = temp(:,3);
n = temp(:,4);

tau = 647.096/T;

% To solve for the properties in Region three, the density must be
known.
```

```
% Since we know the pressure and temperature, the density can be
solved for.
delnew = 1;
del = 100;
while abs(del-delnew) > 1e-5
   del = delnew;
   phi_d = n(1)/del;
   for i = 2:40
      phi_d = phi_d + n(i)*I(i)*(del^(I(i)-1))*tau^(J(i));
   end
   rho = (p*1000)/(R*T*del*phi_d);
   delnew = rho/322;
end
del = delnew;

phi = n(1)*log(del);
for i = 2:40
   phi = phi + n(i)*(del^I(i))*(tau^J(i));
end

phid = n(1)/del;
for i = 2:40
   phid = phid + n(i)*I(i)*(del^(I(i)-1))*tau^(J(i));
end

phidd = -1*(n(1))/(del^2);
for i = 2:40
   phidd = phidd + n(i)*I(i)*del^(I(i)-1)*tau^J(i);
end

phit = 0;
for i = 2:40
   phit = phit + n(i)*del^I(i)*J(i)*tau^(J(i)-1);
end

phitt = 0;
for i = 2:40
   phitt = phitt + n(i)*(del^(I(i)))*J(i)*(J(i)-1)*tau^(J(i)-2);
end

phidt = 0;
for i = 2:40
   phidt = phidt + n(i)*I(i)*del^(I(i)-1)*J(i)*tau^(J(i)-1);
end
```

```
function [eta] = mu(T,P)
% This code calculates the dynamic viscosity of water based on
% the IAPWS equations.
%
% Important variables:
% eta is dynamic viscosity
% Y is reduced dynamic viscosity
% del is reduced density
% tau is inverse reduced temperature
% Input values:
% T in K
% P in Pa
% Output values:
```

```
% eta in Pa.s

etastar = 55.071e-6; %  Pa.s
Tstar = 647.226;     %  K
rhostar = 317.763;   %  kg/m^3

PS = exp(23.196452-3816.44/(T-46.13));
if PS == P
   disp('in vapor region.  Outside of bounds of code');
   quit;
elseif P > PS
   % Pressure is greater than saturation pressure, thus compressed
   [V,H,S]=compressed(T-273.15,P/10^5);
elseif P < PS
   %  Pressure is less than saturation pressure, thus steam
   [V,H,S]=superheat(T-273.15,P/10^5);
end

rho = 1/V;
del = rho/rhostar;
tau = Tstar/T;

%  Ideal gas part
n0 = [.1e1 .978197 .579829 -.202354];
sum = 0;
for i = 1:4
   sum = sum + (n0(i))*tau^(i-1);
end
Y0 = ((tau^.5)*(sum))^-1;

%  Real fluid part
I = [0 0 0 0 1 1 1 1 2 2 2 3 3 3 3 4 4 5 6];
J = [0 1 4 5 0 1 2 3 0 1 2 0 1 2 3 0 3 1 3];
n = [.5132047 .3205656 -.7782567 .1885447 .2151778 ...
      .7317883 .1241044e1 .1476783e1 -.2888107 -.1070786e1...
      -.1263184e1 .1778064 .4605040 .2340379 -.4924179 ...
      -.4176610e-1 .1600435 -.1578386e-1 -.3629481e-2];
sum = 0;
for i = 1:19
   sum = sum + n(i) * (del - 1)^I(i) * (tau - 1)^J(i);
end
Y1 = exp(del * sum);

Y = Y0*Y1;
eta = etastar*Y;
```

```
function [st] = surftens(T)
%  Function is from IAPWS equations and is valid from triple
%  point to critical point.
%
%  Input value:
%  T in K
%  Output value:
%  st in N/m
theta = T/647.096;
st = 235.8*((1-theta)^1.256)*(1-0.625*(1-theta));
st = st/1000;  %  now the units are N/m
```

```
function ps = watersatpres(T)
%  This relationship calculates the saturation pressure for water at a
%  given temperature over the following range:
%  273.15 K < T < 647.096 K or 611.213 Pa < p < 22.064 MPa
%
%  Note:
%  Temperature (T) in K and saturation pressure (ps) in MPa
ns = [0.11670521452767E+04
-0.72421316703206E+06
-0.17073846940092E+02
0.12020824702470E+05
-0.32325550322333E+07
0.14915108613530E+02
-0.48232657361591E+04
0.40511340542057E+06
-0.23855557567849E+00
0.65017534844798E+03];
%  Vapor-Liquid Saturation Curve equations
nu = T + ns(9)/(T - ns(10));
A = nu^2 + ns(1)*nu + ns(2);
B = ns(3)*nu^2 + ns(4)*nu + ns(5);
C = ns(6)*nu^2 + ns(7)*nu + ns(8);
ps = ((2*C)/((-B+sqrt(B^2-4*A*C))))^4;
```

```
function Ts = watersatpres(p)
%  This relationship calculates the saturation temperature for water at
a
%  given pressure over the following range:
%  273.15 K < T < 647.096 K or 611.213 Pa < p < 22.064 MPa
%
%  Note:
%  Temperature (T) in K and saturation pressure (ps) in MPa

ns = [0.11670521452767E+04
-0.72421316703206E+06
-0.17073846940092E+02
0.12020824702470E+05
-0.32325550322333E+07
0.14915108613530E+02
-0.48232657361591E+04
0.40511340542057E+06
-0.23855557567849E+00
0.65017534844798E+03];
%  Vapor-Liquid Saturation Curve equations
beta = p^.25;
E = beta^2 + beta*ns(3) + ns(6);
F = (beta^2)*ns(1) + beta*ns(4) + ns(7);
G = (beta^2)*ns(2) + beta*ns(5) + ns(8);
D = (2*G)/(-1*F - sqrt(F^2 - 4*E*G));
Ts = .5*( ns(10) + D - sqrt( (ns(10)+D)^2 - 4*(ns(9) + ns(10)*D) ) );
```

4. Properties of gas mixtures, including psychrometric relationships

Other codes developed for the analysis of thermodynamic or psychrometric properties are given below:

Subroutines:

1. mix.m – Used to evaluate thermal conductivity or viscosity data for a gas mixture
2. psychrometric.m – Gives psychrometric chart data based on relative humidity and dry bulb temperature

```
function ans = mix(properties,y,M,visc,n)
%  This function calculates the properties of a gas mixture
%  given an array of properties (k or mu/viscosity) where y are the
%  mole fractions, M are the Molecular weights, visc are the
%  viscosities, and n is the number of components in the mixture.
%
%  The web page I found this method on referenced BS&L as the
%  source of this equation for low density gas mixtures.

ans = 0;
for i = 1:n
   a = 0;
   b = 0;
   for j = 1:n
      p1 = 8*(1+M(i)/M(j));
      p2 = (visc(i)/visc(j))^.5;
      p3 = (M(j)/M(i))^.25;
      phi(i,j) = (1/sqrt(p1))*(1+p2*p3)^2;
      b = b + y(j)*phi(i,j);
   end
   a = y(i)*properties(i);
   ans = ans + a/b;
end
```

```
function [Twb,h,omega,specvol] = psychrometric(Tdb,phi)
%  This function provides the data on a psychrometric
%  chart based on the inputs Tdb and phi.
%
%  Input values:
%  Tdb -- Dry bulb temperature, K
%  phi -- Relative humidity, decimal
%
%  Output values:
%  Twb -- Wet bulb temperature, K
%  h -- specific enthalpy, kJ/kg.K
%  omega -- humidity, kg water/kg dry air
%  specvol -- specific volume, m^3/kg

cpa = 1.005;    %  kJ/kg.K
cpw = 4.217;    %  kJ/kg.K
[Vg,Hg,Sg,Pg]=saturatedsteam(Tdb-273.15);
[Vw,Hw,Sw,Pw]=saturatedwater(Tdb-273.15);
pv = phi * Pg;
p = 101325; %  1 atm = 101325 Pa
omega = .622*(pv/(p-pv));
```

```
h = (Tdb - 273.15)*cpa + omega*(cpw*(Tdb-273.15) + (Hg-Hw)/1000);
specvol = (287*Tdb)/(p-pv);

Ta = 0;
Tb = Tdb - 273.15;

[Vgx,Hgx,Sgx,Pgx]=saturatedsteam(Tb);
[Vwx,Hwx,Swx,Pwx]=saturatedwater(Tb);
wb = .622*(Pgx/(p-Pgx));
hb = cpa*Tb + wb*cpw*Tb + wb*(Hg-Hw)/1000;

[Vgx,ha,Sgx,Pgx]=saturatedsteam(Ta);
[Vwx,Hwx,Swx,Pwx]=saturatedwater(Ta);
wa = .622*(Pgx/(p-Pgx));
ha = (wa*ha)/1000;

done = 0;
while done ~= 1;
   if abs(Ta - Tb) < 1e-2;
      done = 1;
   else
      Twb = interp1([ha,hb],[Ta,Tb],h);
      [Vgx,Hgx,Sgx,Pgx]=saturatedsteam(Twb);
      [Vwx,Hwx,Swx,Pwx]=saturatedwater(Twb);
      wwb = .622*(Pgx/(p-Pgx));
      hwb = cpa*Twb + wwb*cpw*Twb + wwb*(Hg-Hw)/1000;
      if hwb > h
         Tb = Twb;
         hb = hwb;
      else
         Ta = Twb;
         ha = hwb;
      end
   end
end

Twb = Twb + 273.15;
```

5. Design of system components

Several MATLAB codes were developed to analyze the different components of the cycle. These are given as follows:

1. vertseparator.m – Analyze and size a vertical separator
2. horizseparator.m – Analyze and size a horizontal separator
3. pump.m – Analyze a pump
4. turbine.m – Analyze a turbine
5. condenser.m – Analyze and size a condeneser
6. coolingtower.m – Analyze and size a cooling tower
7. height.m – Analyze the vertical height for our condenser and turbine
8. pipesize.m – Size piping based on maximum velocity and pressure drop
9. modpipesize.m – Size piping based on maximum velocity and pressure drop, including a change in height term for pressure drop
10. ejector.m – Size an ejector

```
function [f,Sl,Tflash,Dia,H1,H2,H3,H4] =
vertseparator(f,Sf,pflash,Tpond,mdotF)
%  This calculates the flash fraction for the flash distillation
%  procedure from the input initial guess for the flash fraction
%  value.
%  Important Variables:
%  Sf--Salinity of Feed, gm/kg
%  Sl--Salinity of Liquid in flash drum, gm/kg
%  f--Flash fraction
%  pflash--flash pressure, Pa
%  Tpond--flash temperature,K
%  Dia -- Diameter of the flash drum, m
%  H1 -- Height of liquid, m
%  H2 -- Height from liquid to feed inlet, m
%  H3 -- Height from feed inlet to Demister, m
%  H4 -- Height from demister pad to vapor outlet, m
tolerance = 100;

while tolerance >1e-5
%  Calculate Tflash, Kelvin
%  Tsat is in Kelvin, psat in N/m^2
psat=pflash;
Tsat=-3816.44*(log(psat)-23.196452)^(-1)+46.13;
B=(6.71+6.43e-2*(Tsat-273)+9.74e-5*(Tsat-273)^2)*10^-3;
C=(22.238+9.59e-3*(Tsat-273)+9.42e-5*(Tsat-273)^2)*10^-5;
vs=Sf*(B+C*Sf);
Tflash=Tsat+vs;

%  Calculate hvw=hvw(pflash,Tflash)
B1=(Tflash^-1)*2641.62*10^(80870*(Tflash^-2));
B2=82.546;
B3=162460/Tflash;
B4=0.21828*Tflash;
B5=126970/Tflash;
B0=1.89-B1;
Fo=1.89-B1*(372420*Tflash^(-2)+2);
B6=B0*B3-2*Fo*(B2-B3);
```

```
B7=2*Fo*(B4-B5)-B0*B5;
F=1804036.3+1472.265*Tflash+0.37789824*Tflash^2 ...
   +47845.137*log(Tflash);
a=(pflash/(101325*Tflash))^2;
hvw=F+101.31358*(Fo*(pflash/101325)+...
   (.5*B0*a)*(-1*B6+B0*(B2-B3+B0*B7*a)));

%  Calculate hfmix=hfmix(Tpond,Sf)
T=Tpond-273.15;
S=Sf;
A=4206.8-6.6197*S+1.2288e-2*(S^2);
B=-1.1262+5.4178e-2*S-2.2719e-4*(S^2);
C=1.2026e-2-5.3566e-4*S+1.8906e-6*(S^2);
D=6.8774e-7+1.517e-6*S-4.4268e-9*S^2;
h0=2.3e-3*S-1.03e-4*S^2;
h=h0+2.38846e-4*(A*T+.5*B*T^2+(C/3)*T^3+.25*D*T^4);
%  Convert from kcal/kg to J/kg, 1 cal=4.1868 J
hfmix=h*4186.8;

%  Calculate Sl=Sl(Sf,f)
Sl=Sf/(1-f);

%  Calculate hlmix=hlmix(Tflash,Sl)
T=Tflash-273.15;
S=Sl;
A=4206.8-6.6197*S+1.2288e-2*(S^2);
B=-1.1262+5.4178e-2*S-2.2719e-4*(S^2);
C=1.2026e-2-5.3566e-4*S+1.8906e-6*(S^2);
D=6.8774e-7+1.517e-6*S-4.4268e-9*S^2;
h0=2.3e-3*S-1.03e-4*S^2;
h=h0+2.38846e-4*(A*T+.5*B*T^2+(C/3)*T^3+.25*D*T^4);
%  Convert from kcal/kg to J/kg, 1 cal=4.1868 J
hlmix=h*4186.8;

%  Calculate newf
newf = (hfmix-hlmix)/(hvw-hlmix);

%  Are we done yet?
tolerance = abs(newf-f);
f = newf;
end

mdotV = f*mdotF;  %  mass flow rate of vapor leaving separator
mdotL = mdotF-mdotV; %  mass flow rate of liquid leaving separator
[V_V,H_V,S_V]=superheat(Tflash-273.15,pflash/10^5);
rhoV = 1/V_V;
rhoL = 1000*seadensity(Sl,Tflash - 273.15); %   1000*gm/cm^3 = kg/m^3
setvel = 0.07*sqrt((rhoL-rhoV)/rhoV);  %  settling velocity of liquid,
m/s
%  We will assume that a demister pad will be used.
QV = mdotV/rhoV;  %  Vapor Volumetric Flow Rate, m^3/s
Dia = sqrt((4*QV)/(pi*setvel));  %  Minimum diameter of vessel, m
Dia = Dia*2;   %  Let's try a larger separator
if Dia < 2
   Dia = 2;
end
QL = mdotL/rhoL;  %  Liquid Volumetric Flow Rate, m^3/s
```

```
%  We will assume 10 min (600 s) of liquid hold-up time
H1 = (QL*600)/(.25*pi*Dia^2);
if Dia/2 < .6
   H2 = .6;
else
   H2 = Dia/2;
end
if Dia < 1
   H3 = 1;
else
   H3 = Dia;
end
%  Typical demister pads are 100 mm thick.  If ours is considerably
%  thicker, H4 will need to be adjusted.
H4 = .4;
```

```
function [f,Sl,Tflash,Dia,H1,H2,H3,H4] =
horizseparator(f,Sf,pflash,Tpond,mdotF)

%  This function calculates the flash fraction for the flash
%  distillation procedure from the input initial guess for the flash
%  fraction value.
%
%  Key variables:
%  Sf--Salinity of Feed, gm/kg
%  Sl--Salinity of Liquid in flash drum, gm/kg
%  f--Flash fraction
%  pflash--flash pressure, Pa
%  Tpond--flash temperature,K
%  Dia -- Diameter of the flash drum, m
%  H1 -- Height of liquid, m
%  H2 -- Length of Separator, m
%  H3 -- 0 (variable left over from previous code)
%  H4 -- 0 (variable left over from previous code)
tolerance = 100;

while tolerance >1e-5
%  Calculate Tflash, Kelvin
%  Tsat is in Kelvin, psat in N/m^2
psat=pflash;
Tsat=-3816.44*(log(psat)-23.196452)^(-1)+46.13;
B=(6.71+6.43e-2*(Tsat-273)+9.74e-5*(Tsat-273)^2)*10^-3;
C=(22.238+9.59e-3*(Tsat-273)+9.42e-5*(Tsat-273)^2)*10^-5;
vs=Sf*(B+C*Sf);
Tflash=Tsat+vs;

%  Calculate hvw=hvw(pflash,Tflash)
B1=(Tflash^-1)*2641.62*10^(80870*(Tflash^-2));
B2=82.546;
B3=162460/Tflash;
B4=0.21828*Tflash;
B5=126970/Tflash;
B0=1.89-B1;
Fo=1.89-B1*(372420*Tflash^(-2)+2);
B6=B0*B3-2*Fo*(B2-B3);
B7=2*Fo*(B4-B5)-B0*B5;
F=1804036.3+1472.265*Tflash+0.37789824*Tflash^2 ...
   +47845.137*log(Tflash);
```

```
a=(pflash/(101325*Tflash))^2;
hvw=F+101.31358*(Fo*(pflash/101325)+...
   (.5*B0*a)*(-1*B6+B0*(B2-B3+B0*B7*a)));

%  Calculate hfmix=hfmix(Tpond,Sf)
T=Tpond-273.15;
S=Sf;
A=4206.8-6.6197*S+1.2288e-2*(S^2);
B=-1.1262+5.4178e-2*S-2.2719e-4*(S^2);
C=1.2026e-2-5.3566e-4*S+1.8906e-6*(S^2);
D=6.8774e-7+1.517e-6*S-4.4268e-9*S^2;
h0=2.3e-3*S-1.03e-4*S^2;
h=h0+2.38846e-4*(A*T+.5*B*T^2+(C/3)*T^3+.25*D*T^4);
%  Convert from kcal/kg to J/kg, 1 cal=4.1868 J
hfmix=h*4186.8;

%  Calculate Sl=Sl(Sf,f)
Sl=Sf/(1-f);

%  Calculate hlmix=hlmix(Tflash,Sl)
T=Tflash-273.15;
S=Sl;
A=4206.8-6.6197*S+1.2288e-2*(S^2);
B=-1.1262+5.4178e-2*S-2.2719e-4*(S^2);
C=1.2026e-2-5.3566e-4*S+1.8906e-6*(S^2);
D=6.8774e-7+1.517e-6*S-4.4268e-9*S^2;
h0=2.3e-3*S-1.03e-4*S^2;
h=h0+2.38846e-4*(A*T+.5*B*T^2+(C/3)*T^3+.25*D*T^4);
%  Convert from kcal/kg to J/kg, 1 cal=4.1868 J
hlmix=h*4186.8;

%  Calculate newf
newf = (hfmix-hlmix)/(hvw-hlmix);

%  Are we done yet?
tolerance = abs(newf-f);
f = newf;
end

mdotV = f*mdotF;  %  mass flow rate of vapor leaving separator
mdotL = mdotF-mdotV; %  mass flow rate of liquid leaving separator
[V_V,H_V,S_V]=superheat(Tflash-273.15,pflash/10^5);
rhoV = 1/V_V;
rhoL = 1000*seadensity(Sl,Tflash - 273.15); %   1000*gm/cm^3 = kg/m^3
setvel = 0.07*sqrt((rhoL-rhoV)/rhoV);  %  settling velocity of liquid,
m/s
%  We will assume that a demister pad will be used.
QV = mdotV/rhoV;  %  Vapor Volumetric Flow Rate, m^3/s
QL = mdotL/rhoL;  %  Liquid Volumetric Flow Rate, m^3/s

Dv1 = sqrt(QV/(.75*setvel*pi));  % Minimum diameter: vapor velocity

Dv2 = ((8*600*QL)/(3*pi))^(1/3); %  Minimum diameter: liquid hold time
of 10 min

Dia = max([Dv1,Dv2]);
H1 = .5*Dia;
```

```
H2 = 3*Dia;
H3 = 0;
H4 = 0;
```

```
function [power,Q,Head]=pump(mdot,rho,mu,Pin,Pout,Din,Dout,h,eff_p)
%  This function calculates the power needed by the pump.
%
%  Input values:
%  mdot -- mass flow rate, kg/s
%  rho -- density of fluid, kg/m^3
%  mu -- dynamic viscosity in N.s/m^2
%  Pin,Pout -- Inlet and exit pressure
%  Din,Dout -- Inlet and exit diameters
%  h -- height difference between inlet and exit
%  eff_p -- efficiency of pump
%  Output values:
%  power -- power needed by the pump, W
%  Q -- Volumetric flow rate, m^3/s

gamma = 9.81*rho;

%  We will need to loop to assign D3 and D4 to assure laminar flow into
pond
%  Is it possible to assume negligible change in velocity?  Yes!
Q = mdot/rho;
Velin = Q / (pi*.25*Din^2);
Velout = Q / (pi*.25*Dout^2);
Hv = (1/(2*9.81))*(Velout^2 - Velin^2);
Hp = (1/gamma)*(Pout - Pin);
Hz = h;

%  To calculate the losses in the pipes, we will assume a pipe
%  lenth of 1000 m
len = 1000;
Re = Dout*Velout*rho/mu;
if Re > 1e8
   disp(sprintf('Pump Reynold number out of range, Re = %f \n',Re))
end

if Re < 2500
   f = 64/Re;
else
   %  Iterate on the Colebrook formula for friction factor
   fnew = .02;
   f = 1;
   while abs(f-fnew) > 1e-5
      f = fnew;
      epsilon = .045e-3;   %  equivalent roughness of steel, m
      a = -2.0*log10( (epsilon/Dout)/3.7 + 2.51/(Re*sqrt(f)));
      fnew = (a^-1)^2;
   end
   f = fnew;
end
loss = f*(len/Dout)*.5*(Velout^2/9.81);   %  Losses are frictional
losses in pipes

H = Hv + Hp + Hz + loss;   % Total losses, m
output = Q*gamma*H;
```

```
power = output/eff_p;

Head = H;
```

```
function [Tout,pout,wt,x2] = turbine(Tin,pin,x,eff,mdot);
%  This code calculates the power and exit state of the
%  the turbine.
%
%  Input values:
%  Tin -- inlet temperature, K
%  pin -- inlet pressure, Pa
%  x -- estimated exit quality desired (isentropic)
%  eff -- isentropic efficiency of turbine
%  mdot -- mass flow rate of the steam, kg/s
%  Output values:
%  Tout -- outlet temperature of the steam, K
%  pout -- outlet pressure of the steam, Pa
%  wt -- Estimated power generated by the turbine, W
%  x2 -- Estimated quality at turbine exit

%  To fix the isentropic state, we need to find where S1 = S2
%  Our initial guess is at the same temperature as the flow entering
%  the turbine.
[V1,H1,S1] = superheat(Tin-273.15,pin/10^5);

%  We need a way to work our way in to the appropriate value for
%  the exit temperature and pressure of the turbine operating
%  isentropically.
%  Let's use the flash temperature as an initial guess on one side
%  and T = 0 C on the other, then we can use the bisection method
%  to zero in on the right value
Ta = Tin - 273.15;
Tb = 0;
done = 0;
while done ~= 1
   [V2a,H2a,S2a,P2a] = quality(Ta,x);
   [V2b,H2b,S2b,P2b] = quality(Tb,x);
   Si = [S2a,S2b];
   Ti = [Ta,Tb];
   Tnew = interp1(Si,Ti,S1);
   [V2new,H2new,S2new,P2new] = quality(Tnew,x);

   if S2a < S1
      if S2new < S1
         Ta = Tnew;
      else
         Tb = Tnew;
      end
   else
      if S2new > S1
         Ta = Tnew;
      else
         Tb = Tnew;
      end
   end

   if(S2new-S1) < 1e-5
      done = 1;
```

```
   end

end

% Isentropic properties
T2s = Tnew; %  deg. C
[V2s,H2s,S2s,P2s] = quality(T2s,x);
% Now, we know the turbine efficiency and the turbine power, so we can
find
% the actual exit state.
H2 = H1 - eff*(H1 - H2s);
if abs(H2s - H2) > 1e-5
   if H2 < H2s
      disp('Error. Violation of Second Law in turbine.')
      quit;
   else
      % assuming we are still in the 2 phase region, the temperature
and
      %  pressure should be the same as the isentropic temperature and
pressure.
      P2 = P2s;
      T2 = T2s;
      [Vg,Hg,Sg,Pg]=saturatedsteam(T2);
      [Vf,Hf,Sf,Pf]=saturatedwater(T2);
      x2 = (H2 - Hf)/(Hg - Hf);
      [V2,H2temp,S2,P2] = quality(T2,x2);
      if x2 > 1
         disp('not in saturated region at turbine exit.');
         %  Since we are out of the 2 phase region, the exit pressure
will
         %  be the same, but the temperature will have changed.  We
will need
         %  to iterate on the properties to find the temperature that
corresponds
         %  to the exit enthalpy.
         Ta = Tin;
         Tb = T2s;
         [Va,Ha,Sa] = superheat(Ta,P2s/10^5);
         [Vb,Hb,Sb] = superheat(Tb,P2s/10^5);
         done = 0;
         while done ~= 1
            Tnew = interp1([Ha,Hb],[Ta,Tb],H2)
            [Vnew,Hnew,Snew] = superheat(Tnew,P2s/10^5);
            if abs(H2-Hnew) < 1e-5
               done = 1;
               T2 = Tnew;
            else

               if Hnew < H2
                  Tb = Tnew;
                  Hb = Hnew;
               else
                  Ta = Tnew;
                  Ha = Hnew;
               end
            end
         end
```

```
      end
   end
end
wt = (H1-H2)*mdot;
Tout = T2 + 273.15;  %  Kelvin
pout = P2;
```

```
function [mdotl,mdotlout,mdottoto,gasoutflow,numbernozzle,...
      L_vessel,final_D,Din] = ...
   condenser(Tvin,Tvout,Tlin,Tlout,mdoti,p,dp,E)
%  This function makes the condenser design
%  calculations based on the direct contact HEX
%  design parameters presented in Azbel.
%
%  The following is a list of variables:
%  Tvin -- Temperature of the inlet vapor, K
%  Tvout -- Temperature of the outlet vapor, K
%  Tlin -- Temperature of the inlet liquid, K
%  Tlout -- Temperature of the outlet liquid, K
%  mdoti -- Flowrate of inlet stream, kg/s
%        --Composition of inlet stream
%  mdoto -- Flowrate of outlet stream, kg/s
%        -- Composition of outlet stream
%  p -- Operating pressure, Pa
%  dp -- Allowable pressure drop, Pa/m
%  deltapn -- Pressure drop in nozzle
%  Physical properties of liquid:
%     cpl -- specific heat, J/kg.K
%     mul -- dynamic viscosity, N.s/m^2
%     kl -- thermal conductivity, W/m.K
%     rhol -- density, kg/m^3
%     sigma -- surface tension, N/m
% Physical properties of vapor:
%     cpv -- specific heat, J/kg.K
%     muv -- viscosity, N.s/m^2
%     kv -- thermal conductivity, W/m.K
%     rhov -- density, kg/m^3
%     nu -- diffusivity, m^2/s
%     lambda -- latent heat, J/kg
% Saturation pressure and temperature for condensing component:
%     psat -- Pa
%     Tsat -- K
% Spray nozzle characteristics:
%     Qs -- flowrate, m^3/s
%     dps -- nozzle pressure drop, Pa
%     mds -- mean drop size, m
% E -- Entrainment allowance, kg liquid entrained/kg vapor vented
%
%  The output parameters are:
%     mdotl -- Flow rate of the cooling liquid into the condenser, kg/s
%     mdotlout -- Flow rate of the liquid out, kg/s
%     mdottoto -- Flow rate of the vapor out, kg/s
%     gasoutflow -- Volumetric flow rate of the vapor out, cfm
%     numbernozzle -- Estimated number of nozzles
%     L_vessel -- Estimated length of the vessel, m
%     final_D -- Estimated diameter of the vessel, m
%     Din -- Estimated diameter of the vapor inlet, m
```

```
%  Physical properties will be determined at the average temperature
Tvavg = .5*(Tvin+Tvout);
Tlavg = .5*(Tlin+Tlout);
%  Physical properties for vapor side
[r,vv,u,h,s,cpv,cv,muv,kinvis,kv] = whichregion(Tvavg,p/10^6);
rhov = 1/vv;
nu = 0.26e-4;  % at 1 atm and 298 K
const = (0.26e-4*1.0133e5)/(298^1.5);  %  Dab <proportional> p^-1*T^1.5
nu = const*(Tvavg^1.5)/p;
[V,Hg,S,P]=saturatedsteam(Tvavg);
[V,Hf,S,P]=saturatedwater(Tvavg);
lambda = Hg-Hf;
%  Physical properties for liquid side
[r,vl,u,h,s,cpl,cv,mul,kinvis,kl] = whichregion(Tlavg,p/10^6);
if ~(r == 1 | r == 3)
   disp('Out of liquid range for liquid in condenser.')
   region = r
end
rhol = 1/vl;
sigma = surftens(Tlavg);
%  Physical properties for inerts (air) -- found at average vapor
temperature
%  How much inert air can we expect?  Figure 3.33a of Azbell gives
%  a linear relationship between steam to condenser in a steam
%  generating plant and air leakage.
mdotinert = 1.025e-4*(mdoti-100)+0.015;
if mdoti > 500
   disp('Warning: Out of range of Figure 3.33a')
end
%  Properties for viscosity and thermal conductivity are from
%  a "Sutherland Liquid Law" approximation.
mui = (1.716e-5)*((273.15+110.6)/(Tvavg+110.6))*(Tvavg/273.15)^1.5;
ki = (2.414e-2)*((273.15+194.4)/(Tvavg+194.4))*(Tvavg/273.15)^1.5;
rhoi = p/(287*Tvavg);   %  Ideal gas law
if Tvavg > 250 & Tvavg < 300
   cpi = interp1([250,300],[1.003,1.005],Tvavg);
else
   cpi = (3.653 - 1.337e-3*Tvavg + 3.294e-6*Tvavg^2 -...
      1.913e-9*Tvavg^3 + 0.2763e-12*Tvavg^4)*8.314/28.97;
end
if Tvavg > 1000 | Tvavg < 250
   disp('cpi out of Range in Condenser Code.')
end
%  Spray Nozzle Calculations
%  We have lots of data on nozzles, but for the purposes of this
%  calculations we will assume we are using:
%  2 1/2 H60 at 20 psi pressure drop for 115 gpm and 82 deg. spray
%  angle
deltapn = 20*6894.75729317;   %  psi to Pa
Qn = 115 * (1/60) * .003785411784;  %  flow rate in m^3/s
mdotn = rhol*Qn;
%  Vapor Composition
pA = watersatpres(Tvout);  % pressure of vapor at vent
pB = p - pA;               % pressure of inerts at vent
moleA = pA/p;  %  Mole Fraction of vapor
moleB = pB/p;  %  Mole Fraction of inerts
```

```
Mair = 28.97;  %  Molecular weight of air, kg/kgmol
Mwater = 18.02;   %  Molecular weight of water, kg/kgmol
moleflowB = mdotinert/Mair;
moleflowA = moleA*(moleflowB/moleB);

mdotvi = mdoti;
mdottoti = mdotvi + mdotinert;
MassfractionAi = mdotvi/mdottoti;
MassfractionBi = mdotinert/mdottoti;

mdotvo = moleflowA*Mwater;
mdottoto = mdotvo + mdotinert;
MassfractionAo = mdotvo/mdottoto;
MassfractionBo = mdotinert/mdottoto;

%  Enthalpy balance
hbal = (mdotvi-mdotvo)*lambda/1000 ...
+ mdotvi*cpv*(Tvin-Tvout)...
+ (mdotvi-mdotvo)*cpl*(Tvout-Tlout)...
+ mdotinert*cpi*(Tvin-Tvout);
mdotl = hbal/(cpl*(Tlout-Tlin));
numbernozzle = ceil(mdotl/mdotn);   % round to next highest integer
%  liquid out:
mdotlout = mdotl + mdotvi - mdotvo;

%  Spray Characteristics
%  Drop Size (using equations 3.98 and 3.100 from Azbel)
F = Qn/sqrt(deltapn);
dropsize = 2.55*(F/deltapn)^(1/3);  % Correlation for water at 20 deg C
if dropsize < 500e-6
   dropsize = 500e-6;
end
%  For the above situation, dropsize = 1329e-6 m.  This causes the code
to crash.
%  We have modified the estimation to 1000e-6 m to prevent this.
dropsize = 1000e-6;

%  Initial drop velocity given by equation 3.102 from Azbel
uLin = 0.8*sqrt(2*deltapn/rhol);

%  Physical properties of vapor/gas mixture:
%  Based on the vapor composition, it is necessary
%  to perform weighted averages of the vapor/air mixture
%  for the calculation of the length of the spray section.
pmix = p;
tempmix = Tvout;
mfair = MassfractionBo;
yair = moleB;
Mmix = (1-yair)*Mwater + yair*Mair;
cpmix = (1-yair)*cpv + yair*cpi;
rhomix = pmix/((8314/Mmix)*(tempmix)); %  Ideal gas law
mumix = mix([mui,muv],[yair,1-yair],[Mair,Mwater],[mui,muv],2);
kmix = mix([ki,kv],[yair,1-yair],[Mair,Mwater],[mui,muv],2);
Prmix = mumix*cpmix/kmix;
Scmix = mumix/(rhomix*nu); %  Schmidt number = mu/(rho*nu)
Bmix = (rhomix/rhol)*(mumix/rhomix)^0.84;
```

```
%  Stepwise calculations
total_length = 0;
droptemp = Tlin;
while droptemp < Tlout
   duL = -.1;  %  This is a guess, hopefully it will
   %  give a reasonable number of increments
   uLbar = uLin + .5*duL;
   deltaz = (uLbar*duL)/(9.81 -
(81/4)*((Bmix*uLbar^1.16)/(dropsize^1.84)));
   deltat = deltaz/uLbar;
   theta_avg = .5*(droptemp + tempmix);
   Re = uLbar*dropsize*rhomix/mumix;
   hmix = (kmix/dropsize)*(2 + .55*sqrt(Re)*Prmix^(1/3));
   pBi = p;
   pBbar = (pBi - pB)/log(pBi/pB);
   kGMw = (nu*p*Mwater)*(2 + .55*sqrt(Re)*Scmix^(1/3))/...
      (dropsize*8314.4*theta_avg*pBbar);
   qs = hmix*(tempmix - droptemp);
   pA1 = watersatpres(tempmix);
   pAi = watersatpres(Tlin);
   mlambda = kGMw*(pA1 - pAi)*lambda;  %  latent heat flux
   q = qs + mlambda; %  Total heat flux

   dthetaL = (6*q*deltat)/(cpl*dropsize*rhol);
   droptemp = droptemp + dthetaL;
   uLin = uLin + duL;
   total_length = total_length + deltaz;
end
L_vessel = total_length + 0.3;   %  Add length for disengaging height

%  Determination of Diameters:
%  Inlet pipe size
Din = .1;
done = 0;
while done ~= 1
   uGin = mdottoti/(.25*pi*Din^2*rhol);
   Rep = uGin*rhol*Din/mul;
   if Rep > 1e8
      disp(sprintf('Condenser Reynold number out of range, Rep = %f
\n',Rep))
   end
   if Rep < 2500
      f = 64/Rep;
   else
      %  Iterate on the Colebrook formula for friction factor
      fnew = .02;
      f = 1;
      while abs(f-fnew) > 1e-5
         f = fnew;
         epsilon = .045e-3;   %  equivalent roughness of steel, m
         a = -2.0*log10( (epsilon/Din)/3.7 + 2.51/(Rep*sqrt(f)));
         fnew = (a^-1)^2;
      end
      f = fnew;
   end
   dp_z = 2*f*rhol*uGin^2/Din;
   if dp_z < dp
```

```
      done = 1;
   else
      Din = Din*1.1;
   end
end
% Vessel diameter
   K_E = 1.77*(E^(1/3))*((9.81*sigma)/(rhol-rhomix))^.25;
   uGmax = K_E*((rhol-rhomix)/rhomix)^.5;
   temp = mdottoto/(rhomix*0.112);  % Assumes area for entrainment
0.112D^2
   Dvesmin = sqrt(temp/uGmax);
   Dves = 1.25*Din;
   if Dves > Dvesmin
      final_D = Dves;
   else
      final_D = 1.1*Dvesmin;
   end


   gasoutflow = mdottoto/rhomix; %  m^3/s
   gasoutflow = gasoutflow * 60 * 35.3146667215;   % convert to cfm
```

```
function [Im,Taout,L,G,LoverG] = coolingtower(Tdb,phi,Twin,Twout,mdotw)
% This function calculates the design parameters of a cooling
% tower.
%
% A few key variables are:
% Tdb -- dry bulb temperature of inlet air, K
% Tdp -- dew point temperature of inlet air, K
% phi -- relative humidity
% p -- pressure of inlet air, Pa
% cpa -- specific heat of inlet air, kJ/kg.K
% Twin -- water inlet temperature, K
% Twout -- water exit temperature, K
% mdotw -- mass flow rate of the water, kg/s
%
% Im -- Merkel Integral
% Taout -- Air Temperature leaving the cooling tower, K
% L -- Liquid flow rate, kg/s
% G -- Gas flow rate,kg/s
% LoverG -- ratio of liquid flow rate to gas flow rate

[Twb,hairin,omega,specvol] = psychrometric(Tdb,phi);
cpa = 1.005;   % kJ/kg.K
cpw = 4.217;   % kJ/kg.K
p = 101325;    % 1 atm = 101325 Pa

if Twb > Twout
   disp('Error.  Theoretically impossible to cool below wet bulb
temperature in cooling tower.')
end

[trash,hairoutmax,trash,trash] = psychrometric(Tdb,1.0);
LoverGmax = (1/cpw)*((hairoutmax - hairin)/(Twin - Twb));
LoverG = .5*LoverGmax;  % This is something we can play with.

Tain = Tdb;
Taout = (LoverG)*(cpw/cpa)*(Twin - Twout) + Tain;
```

```
% This analysis is for a countercurrent type cooling tower.
% Now we can integrate the Merkel Equation for
% KaV/L = Im (or ndu = number of diffusion units)
n = 100; % n must be even for Simpson's method
tempvect = linspace(Twout,Twin,n);
for i = 1:n
   temp = tempvect(i);
   [Vg,Hg,Sg,Pg]=saturatedsteam(temp-273.15);
   [Vw,Hw,Sw,Pw]=saturatedwater(temp-273.15);
   lambda = (Hg-Hw)/1000;
   omega = .622*(Pg/(p-Pg));
   Hs(i) = omega*(cpw*(temp-273.15) + lambda) + cpa*(temp -273.15);
end
hairout = hairin + LoverG*cpw*(Twin - Twout);
H = linspace(hairin,hairout,n);
for i = 1:n
   a(i)= cpw * (Hs(i)-H(i))^-1;
end
% Now to integrate using Simpson's method
Simp = a(1);
for i = 2:2:n-2
   Simp = Simp + 4*a(i) + 2*a(i+1);
end
Simp = Simp + 4*a(n-1);
Simp = Simp + a(n);
block = (Twin - Twout)/(n-1);
Im = Simp*block/3;

L = mdotw;
G = mdotw/LoverG;
```

```
function [hz]=height(P1,T1,P2,Dia,l,mdot)
% This function calculates the height above
% the ground that the condenser/turbine need to be
% to make up for the pressure difference.
%
% Input parameters:
% P1 -- pressure at condenser exit, Pa
% T1 -- temperature at condenser exit, K
% P2 -- pressure at ground/exit level, Pa
% Dia -- diameter of piping at condenser exit, m
% l -- length of piping, m
% mdot -- mass flow rate through pipe, kg/s
%
% Output parameters:
% hz -- height, m

mu_avg = mu(T1,.5*(P1+P2));   % dynamic viscosity
% Changes in pressure have a minimum impact on density of water, so
% the saturated water value should suffice.
[V1,H1,S1,P1]=saturatedwater(T1-273.15);
rho = 1/V1;
Vel = (mdot)/(rho*(Dia^2)*.25*pi);
Re = (Vel * Dia * rho)/(mu_avg);
if Re < 2500
   f = 64/Re;
else
   fnew = .02;
```

```
   f = 1;
   while abs(f-fnew) > 1e-5
      f = fnew;
      epsilon = 0.045e-3;  %  equivalent roughness of steel, m
      a = -2.0*log10( (epsilon/Dia)/3.7 + 2.51/(Re*sqrt(f)));
      fnew = (a^-1)^2;
   end
   f = fnew;
end

hl = f*(1/Dia)*.5*(Vel^2/9.81);
hz = (P2 - P1)*(rho*9.81)^-1 + hl;
```

```
function dia = pipesize(maxvel,maxdp,rho,mu,mdot)
%  This subroutine calculates the recommended pipe diameter
%  for a fluid system.
%  Variables:
%     maxvel -- maximum allowed velocity, m/s
%     maxdp -- maximum allowed pressure drop per pipe length, Pa/m
%     rho -- density of fluid in pipe, kg/m^3
%     mu -- viscosity of fluid in pipe, kg/m.s
%     mdot -- mass flow rate in pipe, kg/s
%     dia -- recommended diameter, m
%     Re -- Reynold's number
%     vel -- current velocity, m/s
%     f -- friction factor
%     dpactual -- actual pressure drop with current velocity and
diameter
%
%  First we will find the recommended diameter based on maximum
%  velocity.  Using that diameter, we will find the head loss/unit
%  length of pipe.  If the head loss exceeds the maximum, we will
%  increase the diameter of the pipe.
done = 0;
dia = 2*sqrt(mdot/(rho*maxvel*pi));
vel = maxvel;

iteration = 0;
while done ~= 1
   iteration = iteration + 1;
   Re = rho*maxvel*dia/mu;
   if Re < 2500
      f = 64/Re;
   else
      fnew = .02;
      f = 1;
      while abs(f-fnew) > 1e-5
         f = fnew;
         epsilon = .045e-3;   %  equivalent roughness of cast iron, m
         a = -2.0*log10( (epsilon/dia)/3.7 + 2.51/(Re*sqrt(f)));
         fnew = (a^-1)^2;
      end
      f = fnew;
   end

   dpactual = (.5*rho*vel^2*f)/dia;
   if dpactual < maxdp
      done = 1;
```

```
      if iteration == 1
         return;
      end
   else
      db = dia;
      dpb = dpactual;
      dia = 1.1*db;
   end
end

da = dia;
dpa = dpactual;
done = 0;
while done ~= 1;
   if abs(dpactual - maxdp) < 1e-5
      done = 1;
   else
      dia = interp1([dpa,dpb],[da,db],maxdp);
      Re = rho*maxvel*dia/mu;
      if Re < 2500
         f = 0.316/Re^.25;
      else
         fnew = .02;
         f = 1;
         while abs(f-fnew) > 1e-5
            f = fnew;
            epsilon = .045e-3;   % equivalent roughness of cast iron, m
            a = -2.0*log10( (epsilon/dia)/3.7 + 2.51/(Re*sqrt(f)));
            fnew = (a^-1)^2;
         end
         f = fnew;
      end

      dpactual = (.5*rho*vel^2*f)/dia;
      if dpactual > dpa
         dpa = dpactual;
         da = dia;
      else
         dpb = dpactual;
         db = dia;
      end
   end
end

if Re > 1e8
   disp('Reynolds number in pipe is high.')
   Re = Re
end
```

```
function dia = modpipesize(maxvel,maxdp,rho,mu,mdot,factor)
%  This subroutine calculates the recommended pipe diameter
%  for a fluid system.
%  Variables:
%     maxvel -- maximum allowed velocity, m/s
%     maxdp -- maximum allowed pressure drop per pipe length, Pa/m
%     rho -- density of fluid in pipe, kg/m^3
%     mu -- viscosity of fluid in pipe, kg/m.s
%     mdot -- mass flow rate in pipe, kg/s
```

```
%     height -- height of exit of pipe above inlet
%     dia -- recommended diameter, m
%     Re -- Reynold's number
%     vel -- current velocity, m/s
%     f -- friction factor
%     dpactual -- actual pressure drop with current velocity and
diameter
%     factor -- +1 if z2 > z1 and -1 if z2 < z1, or adust for (z2-z1)/l
%
%  First we will find the recommended diameter based on maximum
%  velocity.  Using that diameter, we will find the head loss/unit
%  length of pipe.  If the head loss exceeds the maximum, we will
%  increase the diameter of the pipe.
done = 0;
dia = 2*sqrt(mdot/(rho*maxvel*pi));
vel = maxvel;
grav = 9.81;

iteration = 0;
while done ~= 1
   iteration = iteration + 1;
   Re = rho*maxvel*dia/mu;
   if Re < 2500
      f = 64/Re;
   else
      fnew = .02;
      f = 1;
      while abs(f-fnew) > 1e-5
         f = fnew;
         epsilon = .045e-3;    %  equivalent roughness of cast iron, m
         a = -2.0*log10( (epsilon/dia)/3.7 + 2.51/(Re*sqrt(f)));
         fnew = (a^-1)^2;
      end
      f = fnew;
   end

   dpactual = factor*rho*grav + (.5*rho*vel^2*f)/dia;
   if dpactual < maxdp
      done = 1;
      if iteration == 1
         return;
      end
   else
      db = dia;
      dpb = dpactual;
      dia = 1.1*db;
   end
end

da = dia;
dpa = dpactual;
done = 0;
while done ~= 1;
   if abs(dpactual - maxdp) < 1e-5
      done = 1;
   else
      dia = interp1([dpa,dpb],[da,db],maxdp);
```

```
      Re = rho*maxvel*dia/mu;
      if Re < 2500
         f = 0.316/Re^.25;
      else
         fnew = .02;
         f = 1;
         while abs(f-fnew) > 1e-5
            f = fnew;
            epsilon = .045e-3;    %  equivalent roughness of cast iron,
m
            a = -2.0*log10( (epsilon/dia)/3.7 + 2.51/(Re*sqrt(f)));
            fnew = (a^-1)^2;
         end
         f = fnew;
      end

      dpactual = factor*rho*grav + (.5*rho*vel^2*f)/dia;
      if dpactual > dpa
         dpa = dpactual;
         da = dia;
      else
         dpb = dpactual;
         db = dia;
      end
   end
end

if Re > 1e8
   disp('Reynolds number in pipe is high.')
   Re = Re
end
```

```
function [ws,wsGPM] = ejector(mdot, mair, pressure, supress)
%  This function sizes the ejector for our
%  system.
%  ws -- steam consumed, kg/s
%  mdot -- total working fluid handled, kg/s
%  mair -- mass of air/condensables, kg/s
%  pressure -- pressure of the steam, Pa
%  supress -- suction pressure (of condensables), in Hg absolute
pressure = pressure* 1.4503773773e-4;  %  convert to psi
mdot = mdot * 3600 * 2.20462262185; %  convert to lb/hr
mair = mair * 3600 * 2.20462262185; %  conver to lb/hr

if supress < 0.5 | supress > 3.5
   disp('Adjust Suction Pressure')
   supress = supress
end
wsprime = 6.8626*(supress^-.5356);

if mair/(mdot) < .1 | mair/(mdot) > .95
   disp('Wa/Wm is out of range.')
   mair/(mdot)
end
K = .9969*(mair/(mdot))^.4048;

if pressure < 60 | pressure > 200
   disp('Pressure out of range in ejector')
```

```
   pressure = pressure
end
F = 2.55978E-09*(pressure)^4 - 1.54722E-06*(pressure)^3 + ...
   3.50016E-04*(pressure)^2 - 3.64557E-02*pressure + 2.40392E+00;

ws = wsprime*mdot*K*F
wsGPM = ws*.06;
if wsGPM < 10
   wsGPM = 10;
end

ws = ws/(3600*2.20462262185); %  convert to kg/s
```

6. Design of power cycle, and printout of results

The following code combines the other codes to analyze our system. The results are then displayed in the MATLAB command window.

The code and results are given in the following section.

```
function powercycle
%  This code serves as a combining point to analyze/design
%  the solar pond power cycle from different codes to analyze
%  the individual components.'
clear;
clc;
format long g;
format compact;

Tpond = 80 + 273.15;
Tamb = 294.26; % 70 deg. F
Pamb = 1.013e5;   % 1 atm
S_LCZ = 150;
S_UCZ = 50;
LCZ = 2;
NCZ = 2;
UCZ = .5;
d = LCZ + NCZ + UCZ;
eff_t = 0.8;
eff_p = 0.8;
power = 5e6;
mdotF = 6.0e3; %  kg/s flowing out of pond.

done = 0;
while  done ~= 1

   rho_b = 1000*seadensity(S_LCZ,Tpond - 273.15);  % convert to kg/m^3
   rho_w = 1000*seadensity(S_UCZ,Tamb - 273.15);   % convert to kg/m^3
   Ppond = Pamb + 9.81*((UCZ + NCZ)*rho_w + LCZ*rho_b);  % in Pa

   pflash = 28700;    %  in Pa

   %  Flash separator calculation
   [f,Sl,Tflash,Dia,H1,H2,H3,H4]=...
      horizseparator(.5,S_LCZ,pflash,Tpond,mdotF);

   if Sl > 350
      disp('Error.  Salt precipitation in separator');
   end
   %  Throttle calculation
   mu_t = seadynvisc(S_LCZ,Tpond-273.15);
   maxvel = 2;     %  m/s, This number should be between 4 and 10 ft/s
for water
   maxdp = 500;       %  Pa/m
   Dt = pipesize(maxvel,maxdp,rho_b,mu_t,mdotF);
   if Dt < .25
      Dt = .25;
   end
```

```
   throtheight = .5*(Pamb-pflash)/(rho_b*9.81);
   Pthrot = Pamb - throtheight*rho_b*9.81;
   Area = .25 * pi * (Dt)^2;
   Kl = (2*rho_b*(Pthrot - pflash)*Area^2)/(mdotF^2);
   Re_t = (mdotF*Dt)/(Area*mu_t);
   if Re_t > 10^8
      disp(sprintf('Evaluate throttle pipe diameter, Re_t =
%f\n',Re_t))
   end

   % Mass flow rates leaving separator, kg/s
   mdotP = mdotF*(1-f);

   % Pump power and volumetric flow rate calculation
   rho_P = 1000*seadensity(Sl,Tflash-273.15);   % convert to kg/m^3
   mu_P = seadynvisc(Sl,Tflash-273.15);
   Pin_p = pflash;
   Pout_p = Ppond;
   Din = Dt;   % Assume same pipe diameter as for throttle
   Dout = Dt;
   % Net positive suction head available should be calculated in order
   % size the pumps.
   [pwater,psea] = vaporpressure(Sl,Tflash-273.15);
   psea = psea*10^5;
   g = 9.81;
   NPSHA1 = 5; % Assumed depth of pump below separator
   h = (Dia-throtheight-d) + NPSHA1;   % Assumes pump must raise
liquid from inlet height minus
   % the height of the throttle minus the depth of pond plus the depth
of the pump
   Din = pipesize(maxvel,maxdp,rho_P,mu_P,mdotP);
   Dout = Din;

[wp,Q_pump,Head]=pump(mdotP,rho_P,mu_P,Pin_p,Pout_p,Din,Dout,h,eff_p);
   % Let's assume x pumps:
   numberpump = 15;
   Din15 = pipesize(maxvel,maxdp,rho_P,mu_P,mdotP/numberpump);
   Dout = Din;

[wp15,Q_pump15,Head15]=pump(mdotP/numberpump,rho_P,mu_P,Pin_p,Pout_p,Di
n,Dout,h,eff_p);
   wptot = wp15*numberpump;

   % Mass flow rate through turbine
   mdotT = mdotF*f;

   % Turbine power and exit conditions
   [r,vtoT,u,h,s,cp,cv,mutoT,kinvis,k] =
whichregion(Tflash,pflash/10^6);
   Dturb = pipesize(61,.1e5/100,1/vtoT,mutoT,mdotT);
   modDturb = modpipesize(61,.1e5/100,1/vtoT,mutoT,mdotT,1);
   quality = 0.94;
   [Tout,pout,wt,quality2]= turbine(Tflash,pflash,quality,eff_t,mdotT);

   % Condenser design cycle
   dpc = .5;   % Allowable pressure drop, Pa/m
   E = .05;    % Entrainment ratio
```

```
   %  We can play with these temperatures a little bit.
   Tdb = 310.4;    %  99 deg. F.
   phi = .44;
   [Twb,h,omega,specvol] = psychrometric(Tdb,phi);
   Tlin = Twb + 5;    %  Allow a 5 deg. K approach
   if Tlin < 273.15 + 30
      Tlin = 273.15 + 30;  %  30 deg. min.
   end
   Tlout = Tlin + 10;
   Tvin = Tout;
   Tvout = Tvin - 5;

[mdotl,mdotlout,mdottoto,gasoutflow,numbernozzle,L_vessel,final_D,Din]
= ...
      condenser(Tvin,Tvout,Tlin,Tlout,mdotT,pout,dpc,E);
   wmv = 20 * 745.699871582;  %  Estimated power requirement for
mechanical vacuum is 20 hp converted to W

   %  Cooling tower design cycle
   Twin = Tlout;
   Twout = Tlin;
   mvwater = .864;    %  Water for liquid ring vaccuum --> 13.7 gal/min
= 0.864 kg/s
   if mvwater > mdotT
      disp('more water needed for vaccuum than turbine')
   end
   [Im,Taout,L,G,LoverG] =
coolingtower(Tdb,phi,Twin,Twout,mdotl+mvwater);
   [v,h,s] = saturatedwater(Twout - 273.15);
   wct = 450 * 745.699871582; %  Estimated power requirement for
cooling tower is 450 hp converted to W
   volumetric = L * v * 60 / 0.003785411784; %  Gal/min

   %  Height above ground for turbine and condenser
   l = 100;
   [r,vcond,u,h,s,cp,cv,eta_cond,kinvis,k] =
whichregion(Tlout,pout/10^6);
   Dia_c = pipesize(maxvel,maxdp,1/vcond,eta_cond,mdotlout);
   %modDia_c = modpipesize(maxvel,pout/l-
Pamb/l,1/vcond,eta_cond,mdotlout,-1)-->height calculates enough to
handle pressure drop
   hz = height(pout,Tlout,Pamb,Dia_c,l,mdotlout);

   %  Pump power needed to raise cooling water to condenser
   [r,vpump2,u,h,s,cp,cv,eta_pump2,kinvis,k] =
whichregion(Twout,(pout+Pamb)/10^6);
   rho_P = 1/vpump2;
   mu_P = eta_pump2;
   Pin_p = Pamb;
   Pout_p = pout;
   Dia_2 = pipesize(maxvel,maxdp,rho_P,mu_P,mdotl);
   Din = Dia_2;
   Dout = Dia_2;
   h = hz;  %Assumes pump must raise liquid from inlet height

[wp2,Q_pump2,Head2]=pump(mdotl,rho_P,mu_P,Pin_p,Pout_p,Din,Dout,h,eff_p
);
```

```
    pvap = watersatpres(Twout);
    pvap = pvap*10^6;
    Vin_p = Q_pump2/(pi*.25*Din^2);
    g = 9.81;
    h = 0;
    NPSHA2 = Pin_p/(rho_P*g) - h - pvap/(rho_P*g);
    % Let's assume x pumps:
    numberpump10 = 2;
    Din10 = pipesize(maxvel,maxdp,rho_P,mu_P,mdotl/numberpump10);
    Dout = Din;

[wp10,Q_pump10,Head10]=pump(mdotl/numberpump10,rho_P,mu_P,Pin_p,Pout_p,
Din10,Dout,hz,eff_p);

        % enthalpy of pond water entering separator
    Hpond = seaenthalpy(S_LCZ,Tpond - 273.15);
    % enthalpy leaving pump, about to enter pond
    Hlp = seaenthalpy(Sl, Tflash - 273.15);
    % H5 is fluid leaving condenser and entering pond
    [r,v,u,H5,s,cp,cv,eta,kinvis,k] = whichregion(Tlout,Pamb/10^6);
    H5 = H5*1000;
    % Heat input
    q = (Hpond - f*H5 -(1-f)*Hlp)*mdotF;
    % First Law efficiency
    eff = (wt-wptot-wp2-wmv-wct)/q;
    availpower = wt-wptot-wp2-wmv-wct;
    % Fresh water production rate, [(mass fresh water)/(heat input)]
    Rlq = f/q;

    if availpower > 5e6
        done = 1;
    else
        mdotF = mdotF*1.01;
    end

end
% availpower = availpower
% wt = wt
% wp = wp
% wp2 = wp2
% wmv = wmv
% wct = wct

% Pond data
insolation = 20e6;   % 8-31 MJ/m^2-day
pond_eff = .2;
pond_area = (q*(24*3600))/(insolation*pond_eff);
area_ft = pond_area*0.9290304;   % Pond Area in ft^2
nu = 911.42;   % Air velocity, fpm
[trash,prw] = vaporpressure(S_UCZ,40); % Surface temp of water = 104 F
= 40 C
prw = prw / .033863881579; % Convert bar to in. hg
[trash,pra] = vaporpressure(S_UCZ,11.7); % Dewpoint = 53 F = 11.7 C
pra = pra / .033863881579; % Convert bar to in. hg
[trash,Hg,trash,trash]=saturatedsteam(40);
[trash,Hf,trash,trash]=saturatedwater(40);
hfg = Hg - Hf; % J/kg
```

```
hfg = hfg / 2.326;
evaploss = (area_ft*(95 + .425*nu)*(prw - pra))/(hfg);   %  lb/hr
evaploss = evaploss * .45359237; % kg/hr
pond_vol = pond_area*d;
pond_mass = (UCZ/d)*pond_vol*rho_b + ((d-UCZ)/d)*pond_vol*rho_w;
salt_mass = ((UCZ/d)*pond_vol*rho_b*S_UCZ + ((d-
UCZ)/d)*pond_vol*rho_w*S_LCZ)/1000;

%  Present results:
disp('Some significant results of the power cycle design code are as
follows:')
disp('INPUT:')
disp(sprintf('Temperature of pond bottom:\t%f K',Tpond))
disp(sprintf('Pressure at pond bottom:\t%f Pa',Ppond))
disp(sprintf('Ambient Temperature:\t\t%f K',Tamb))
disp(sprintf('Ambient Pressure:\t\t%f Pa',Pamb))
disp(sprintf('Salinity and depth of lower convecting zone:\n\t\t%f
gm/kg\t%f m',S_LCZ,LCZ))
disp(sprintf('Salinity and depth of upper convecting zone:\n\t\t%f
gm/kg\t\t%f m',S_UCZ,UCZ))
disp(sprintf('Total depth of pond:\t\t%f m',d))
disp(sprintf('Efficiency of turbine:\t\t%f',eff_t))
disp(sprintf('Efficiency of pump:\t\t%f',eff_p))
disp(sprintf('Mass flow of brine leaving pond:\t%f kg/s',mdotF))
disp(sprintf('Flash pressure:\t\t\t%f Pa',pflash))
disp(' ')
disp('OUTPUT')
disp('Throttle calculations:')
disp(sprintf('\tDiameter of pipe:\t\t%f m',Dt))
disp(sprintf('\tPressure entering throttle:     %f Pa',Pthrot))
disp(sprintf('\tThrottle coefficient:\t%f',Kl))
disp(sprintf('\tHeight of throttle above pond:     %f m',throtheight))
disp('Separator Calculations:')
disp(sprintf('\tFlash fraction:\t\t\t%f',f))
disp(sprintf('\tSalinity of brine after flashing:\t%f gm/kg',Sl))
disp(sprintf('\tFlash temperature:\t\t%f K',Tflash))
disp(sprintf('\tDiameter of flash drum:\t\t%f m',Dia))
disp(sprintf('\tHeight of Liquid:\t\t%f m',H1))
disp(sprintf('\tLength of Vessel:                 %f m',H2))
disp('Pump 1 Calculations:')
disp(sprintf('\tMass flow through pump:           %f kg/s',mdotP))
disp(sprintf('\tVolumetric flow through pump:     %f m^3/s',Q_pump))
disp(sprintf('\tPumping power required:           %f W',wp))
disp(sprintf('\tNet positive suction head available:     %f m',NPSHA1))
disp(sprintf('\tNecessary head rise:              %f m',Head))
disp('Pump 1 location with x pumps:')
disp(sprintf('\tNumber of pumps:                  %f',numberpump))
disp(sprintf('\tMass flow per pump:               %f kg/s',mdotP/15))
disp(sprintf('\tDiameter of pipe:                 %f m',Din15))
disp(sprintf('\tVolumetric flow per pump:         %f m^3/s',Q_pump15))
disp(sprintf('\tPumping power required, per pump:        %f W',wp15))
disp(sprintf('\tNet positive suction head available:     %f m',NPSHA1))
disp(sprintf('\tNecessary head rise per pump:            %f m',Head15))
disp('Turbine Calculations:')
disp(sprintf('\tMass flow through turbine:       %f kg/s',mdotT))
disp(sprintf('\tTurbine inlet temperature:       %f K',Tflash))
disp(sprintf('\tTurbine inlet pressure:          %f Pa',pflash))
```

```
disp(sprintf('\tPipe diameter to turbine:       %f m',Dturb))
disp(sprintf('\tPipe diameter with height:      %f m',modDturb))
disp(sprintf('\tQuality of Isentropic Process:  %f',quality))
disp(sprintf('\tQuality at turbine exit:        %f',quality2))
disp(sprintf('\tTemperature at turbine exit:    %f K',Tout))
disp(sprintf('\tPressure at turbine exit:       %f Pa',pout))
disp(sprintf('\tTurbine power:                  %f W',wt))
disp('Condenser Calculations:')
disp(sprintf('\tTemperature of vapor in:        %f K',Tvin))
disp(sprintf('\tTemperature of vapor out:       %f K',Tvout))
disp(sprintf('\tTemperature of liquid in:       %f K',Tlin))
disp(sprintf('\tTemperature of liquid out:      %f K',Tlout))
disp(sprintf('\tCondenser operating pressure:   %f Pa',pout))
disp(sprintf('\tAllowable pressure drop:        %f Pa/m',dpc))
disp(sprintf('\tEntrainment:                    %f ',E))
disp(sprintf('\tMass flow rate of vapor in:     %f kg/s',mdotT))
disp(sprintf('\tMass flow rate of vapor out:    %f kg/s',mdottoto))
disp(sprintf('\tMass flow rate of liquid in:    %f kg/s',mdotl))
disp(sprintf('\tMass flow rate of liquid out:   %f kg/s',mdotlout))
disp(sprintf('\tNumber of nozzles:              %f',numbernozzle))
disp(sprintf('\tLength of vessel:               %f m',L_vessel))
disp(sprintf('\tDiameter of vessel:             %f m',final_D))
disp(sprintf('\tDiameter of vapor inlet:        %f m',Din))
disp(sprintf('\tVolumetric Flow Rate of Vapor out:     %f
cfm',gasoutflow))
disp(sprintf('\tEstimated Vacuum Power:         %f W',wmv))
disp('Cooling Tower Specifications:')
disp(sprintf('\tTemperature of liquid in:    %f K, %f F',Twin,
(1.8*(Twin - 273.15) + 32)))
disp(sprintf('\tTemperature of liquid out:   %f K, %f F',Twout,
(1.8*(Twout - 273.15) + 32)))
disp(sprintf('\tTemperature of air in:       %f K, %f F',Tdb, (1.8*(Tdb
- 273.15) + 32)))
disp(sprintf('\tTemperature of air out:      %f K, %f F',Taout,
(1.8*(Taout - 273.15) + 32)))
disp(sprintf('\tWet bulb temperature of inlet air:     %f K, %f F',Twb,
(1.8*(Twb - 273.15) + 32)))
disp(sprintf('\tApproach                     %f K, %f F',Twout -
Twb,(1.8*(Twout - 273.15) + 32 - (1.8*(Twb - 273.15) + 32))))
disp(sprintf('\tCooling Range                %f K, %f F',Twin -
Twout,(1.8*(Twin - 273.15) + 32)-(1.8*(Twout - 273.15) + 32)))
disp(sprintf('\tMass flow rate of water:     %f kg/s',L))
disp(sprintf('\tVolumetric Flow Rate of Water:     %f
gal/min',volumetric))
disp(sprintf('\tMass flow rate of air:       %f kg/s',G))
disp(sprintf('\tL/G                          %f',LoverG))
disp(sprintf('\tKaV/L                        %f',Im))
disp(sprintf('\tEstimated power requirement: %f W',wct))
disp('Pump 2 Calculations:')
disp(sprintf('\tMass flow rate:              %f kg/s',mdotl))
disp(sprintf('\tVolumetric flow rate:        %f m^3/s',Q_pump2))
disp(sprintf('\tDiameter of pipe:            %f m',Dia_2))
disp(sprintf('\tPumping power:               %f W',wp2))
disp(sprintf('\tNecessary head rise:         %f m',Head2))
disp(sprintf('\tNet positive suction head available:     %f m',NPSHA2))
disp('Pump 2 location with x pumps:')
disp(sprintf('\tNumber of pumps:                  %f',numberpump10))
```

```
disp(sprintf('\tMass flow per pump:                 %f
kg/s',mdotl/numberpump10))
disp(sprintf('\tDiameter of pipe:                   %f m',Din10))
disp(sprintf('\tVolumetric flow per pump:           %f m^3/s',Q_pump10))
disp(sprintf('\tPumping power required, per pump:          %f W',wp10))
disp(sprintf('\tNet positive suction head available:       %f m',NPSHA2))
disp(sprintf('\tNecessary head rise per pump:              %f m',Head10))
disp('Other Information:')
disp(sprintf('\tAvailable power:                %f W',availpower))
disp(sprintf('\tHeight of turbine/condenser: %f m',hz))
disp(sprintf('\tPipe Diameter, t/c to ground:      %f m',Dia_c))
disp(sprintf('\tHeat Input:                     %f W',q))
disp(sprintf('\tFirst law efficiency:           %f',eff))
disp(sprintf('\tEstimated Insolation:           %f J/m^2-day',insolation))
disp(sprintf('\tEstimated pond efficiency:      %f',pond_eff))
disp(sprintf('\tEstimated pond area:            %f m^2',pond_area))
disp(sprintf('\tEstimated pond volume:          %f m^3',pond_vol))
disp(sprintf('\tEstimated pond mass:            %f kg',pond_mass))
disp(sprintf('\tEstimated salt mass:            %f kg',salt_mass))
disp(sprintf('\tEstimated evaporation losses:      %f
kg/s',evaploss/3600))
disp(sprintf('\tFresh water generation per heat input:     %f',Rlq))
```

Some significant results of the power cycle design code are as follows:

```
INPUT:
Temperature of pond bottom:      353.150000 K
Pressure at pond bottom:         147951.028487 Pa
Ambient Temperature:             294.260000 K
Ambient Pressure:                101300.000000 Pa
Salinity and depth of lower convecting zone:
       150.000000 gm/kg    2.000000 m
Salinity and depth of upper convecting zone:
       50.000000 gm/kg     0.500000 m
Total depth of pond:              4.500000 m
Efficiency of turbine:            0.800000
Efficiency of pump:               0.800000
Mass flow of brine leaving pond: 6000.000000 kg/s
Flash pressure:                   28700.000000 Pa

OUTPUT
Throttle calculations:
     Diameter of pipe:           1.877837 m
     Pressure entering throttle:      65000.000000 Pa
     Throttle coefficient:        16.755636
     Height of throttle above pond:     3.416032 m
Separator Calculations:
     Flash fraction:              0.007491
     Salinity of brine after flashing:  151.132156 gm/kg
     Flash temperature:           348.232944 K
     Diameter of flash drum:      14.077711 m
     Height of Liquid:            7.038856 m
```

```
     Length of Vessel:                  42.233133 m
Pump 1 Calculations:
     Mass flow through pump:            5955.054744 kg/s
     Volumetric flow through pump:      5.478051 m^3/s
     Pumping power required:            1711344.486056 W
     Net positive suction head available:      5.000000 m
     Necessary head rise:               23.435416 m
Pump 1 location with x pumps:
     Number of pumps:                   15.000000
     Mass flow per pump:                397.003650 kg/s
     Diameter of pipe:                  0.482178 m
     Volumetric flow per pump:          0.365203 m^3/s
     Pumping power required, per pump:         108808.598317
W
     Net positive suction head available:      5.000000 m
     Necessary head rise per pump:             22.350626 m
Turbine Calculations:
     Mass flow through turbine:        44.945256 kg/s
     Turbine inlet temperature:        348.232944 K
     Turbine inlet pressure:           28700.000000 Pa
     Pipe diameter to turbine:         2.285125 m
     Pipe diameter with height:        2.285125 m
     Quality of Isentropic Process:    0.940000
     Quality at turbine exit:          0.956533
     Temperature at turbine exit:      317.880618 K
     Pressure at turbine exit:         9438.444719 Pa
     Turbine power:                    7120859.301970 W
Condenser Calculations:
     Temperature of vapor in:          317.880618 K
     Temperature of vapor out:         312.880618 K
     Temperature of liquid in:         304.853864 K
     Temperature of liquid out:        314.853864 K
     Condenser operating pressure:     9438.444719 Pa
     Allowable pressure drop:          0.500000 Pa/m
     Entrainment:                      0.050000
     Mass flow rate of vapor in:       44.945256 kg/s
     Mass flow rate of vapor out:      0.009357 kg/s
     Mass flow rate of liquid in:      1391.897153 kg/s
     Mass flow rate of liquid out:     1436.842409 kg/s
     Number of nozzles:                194.000000
     Length of vessel:                 2.958902 m
     Diameter of vessel:               1.017534 m
     Diameter of vapor inlet:          0.943634 m
     Volumetric Flow Rate of Vapor out:      188.615731 cfm
     Estimated Vacuum Power:           14913.997432 W
Cooling Tower Specifications:
     Temperature of liquid in:   314.853864 K, 107.066955 F
```

```
    Temperature of liquid out:  304.853864 K, 89.066955 F
    Temperature of air in:      310.400000 K, 99.050000 F
    Temperature of air out:     330.794302 K, 135.759744 F
    Wet bulb temperature of inlet air:     299.853864 K,
80.066955 F
    Approach                    5.000000 K, 9.000000 F
    Cooling Range               10.000000 K, 18.000000 F
    Mass flow rate of water:    1392.761153 kg/s
    Volumetric Flow Rate of Water:    22437.998309
gal/min
    Mass flow rate of air:      2865.532446 kg/s
    L/G                         0.486039
    KaV/L                       0.960089
    Estimated power requirement: 335564.942212 W
Pump 2 Calculations:
    Mass flow rate:             1391.897153 kg/s
    Volumetric flow rate:       1.398709 m^3/s
    Diameter of pipe:           0.943634 m
    Pumping power:              52822.927887 W
    Necessary head rise:        3.094826 m
    Net positive suction head available:    9.897322 m
Pump 2 location with x pumps:
    Number of pumps:                2.000000
    Mass flow per pump:             695.948577 kg/s
    Diameter of pipe:               0.667250 m
    Volumetric flow per pump:       0.699354 m^3/s
    Pumping power required, per pump:       9461.687797 W
    Net positive suction head available:    9.897322 m
    Necessary head rise per pump:           1.108696 m
Other Information:
    Available power:            5085428.459680 W
    Height of turbine/condenser: 10.005553 m
    Pipe Diameter, t/c to ground:    0.960491 m
    Heat Input:                 110681718.908912 W
    First law efficiency:       0.045946
    Estimated Insolation:       20000000.000000 J/m^2-day
    Estimated pond efficiency:  0.200000
    Estimated pond area:        2390725.128433 m^2
    Estimated pond volume:      10758263.077946 m^3
    Estimated pond mass:        11198258941.003599 kg
    Estimated salt mass:        1550255061.368116 kg
    Estimated evaporation losses:    0.225200 kg/s
    Fresh water generation per heat input:    0.000000
```

Appendix 2: Vendor Information

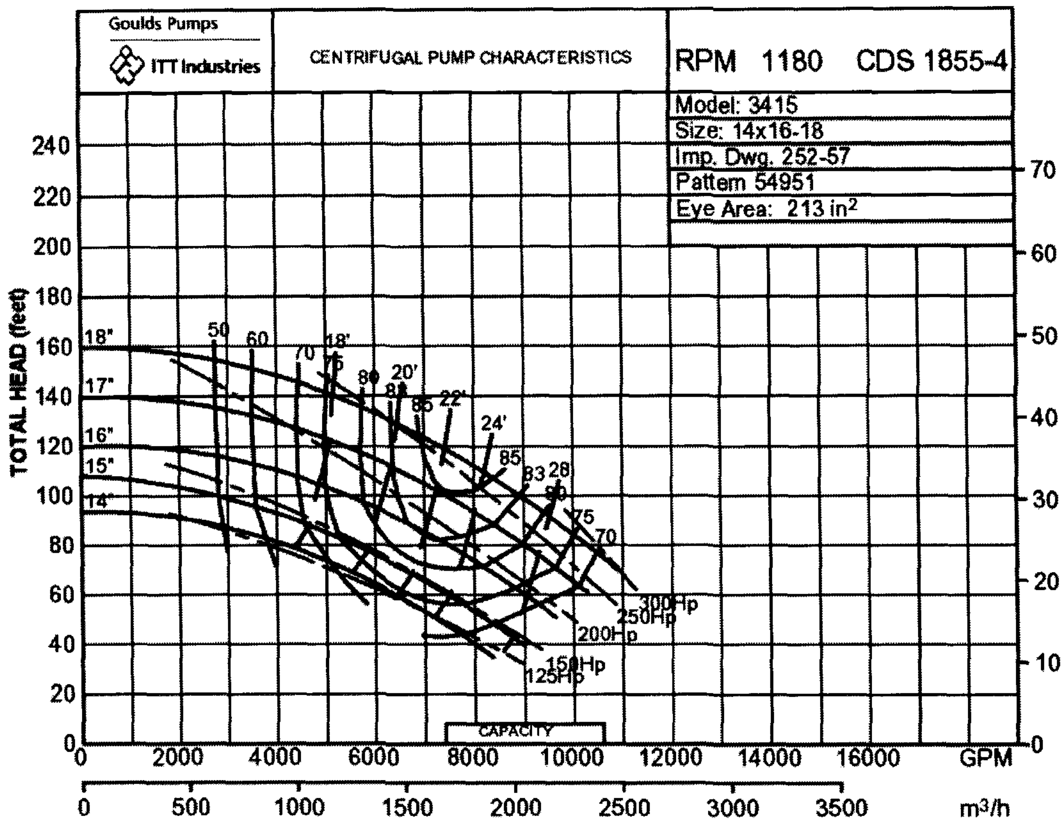


Figure Appendix 2-1: Pump performance curve for Goulds model 3415 14x16-18. Used to return unflashed brine to the pond.

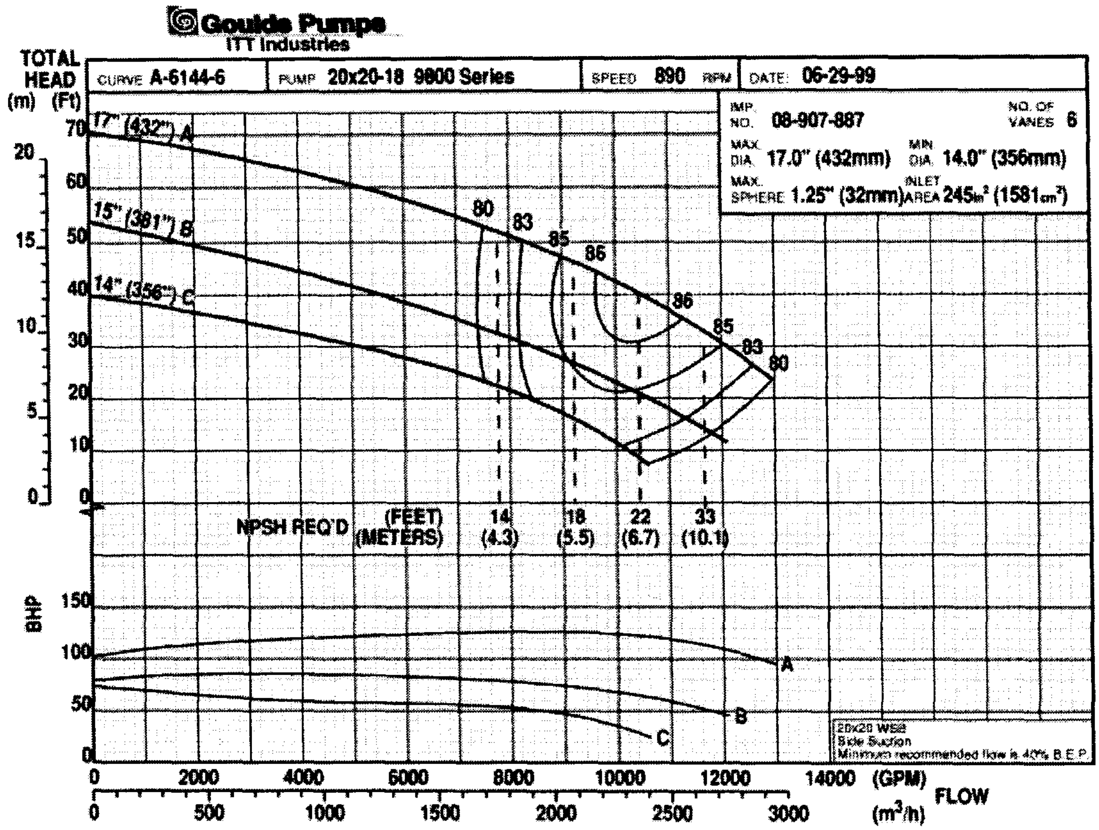


Figure Appendix 2-2: Pump performance curve for Goulds model 3498 20x20-18. This pump will be used to raise the water from the cooling tower to the condenser.

Marley Cooling Tower
A United Dominion Company

Address – City, State, USA zip / Tel: xxx-xxx-xxx / Fax: xxx-xxx-xxxx / E-mail – name@company.com

BUDGETARY SELECTION

TO:	Brett Lee	DATE:	April 12, 2001
ATTN:		FROM:	Duane Thacker

PROJECT: Solar Salt Pond

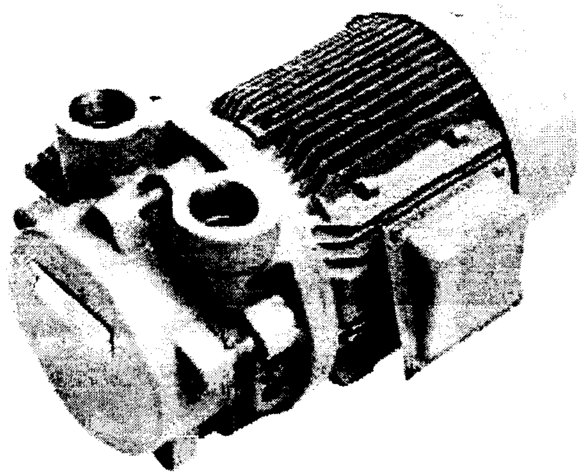
FIELD ERECTED COOLING TOWER

DESIGN CONDITIONS:	Flow	38580 gpm
	Hot Water	107 °F
	Cold Water	89 °F
	Wet Bulb	80 °F
	Plume Abatement	**None**
TOWER DESCRIPTION:	Model	W478-4.0-3
	Number of Cells	3
	Pump Head	20.29 ft
	Fan Diameter	28 ft
	Motor Size	3 @ 150 Hp
	Brake Horsepower	3 @ 148.3 Hp
	Evaporation	638 gpm
	Drift Rate	0.0005 %
TOWER DIMENSION:	Tower Width	42.67 ft
	Tower Length	144.7 ft
	Tower Height	41.79 ft
	Fan Deck Height	28.04 ft
BASIN DIMENSION:	Basin Width	48 ft
	Basin Length	145 ft
BUDGET PRICE:	$588,500 USD	

Marley's budget price is based upon a scope that includes engineering, prefabrication of materials, freight to jobsite and supervision and labor to field assemble the above field erected cooling tower. The following are not included, and should be provided by the purchaser: Concrete cold water basin, anchor bolts, fire protection sprinkler system (if required by Owner's insurance underwriter), pumps, piping, valves, water make-up, motor starter, disconnects, and controls.

Bulletin CHR-1

CHR Series Single Stage Liquid Ring Vacuum Pumps



Rugged and compact pumps for handling suction duties up to 250 cfm

CHR Liquid Ring Vacuum Pumps deliver high performance and high reliability with an advanced design that's very simple and compact. They incorporate the late Stokes technology innovations.

Outstanding features include:

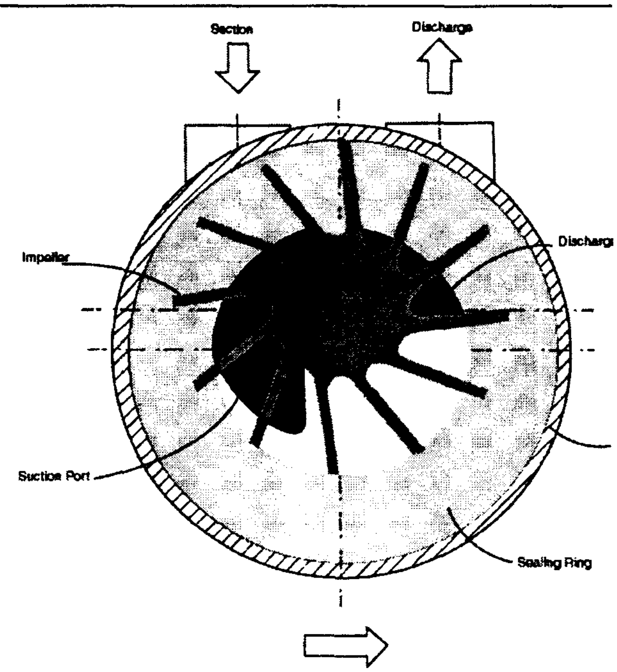
- Simple design, rugged construction
- Minimum maintenance requirements
- Only one simple shaft seal
- Design eliminates misalignment problems
- No bedplate or coupling guard
- Oil-free — no contamination problems
- Handles wet and dust-laden gases
- Low noise level

How it works

The curved-blade impeller is mounted concentrically to the axis of the pump casing, allowing the liquid ring to circulate concentrically within the axis of the casing.

Process gases are drawn through the inlet port into the impeller cells, compressed, then discharged through the outlet port.

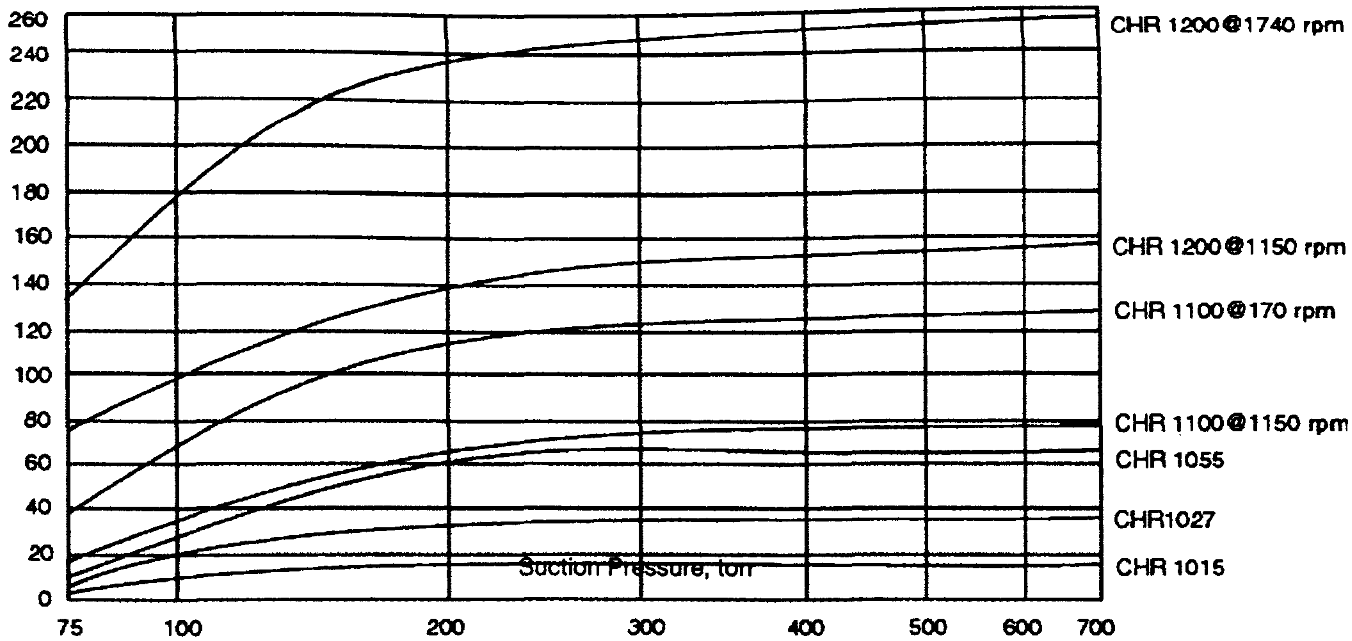
By supplying seal liquid at a pressure equal to the discharge pressure, the pump can automatically make up the amount of liquid eliminated through the outlet port. This allows the heat of compression to be quickly removed and the unit to run cooler.



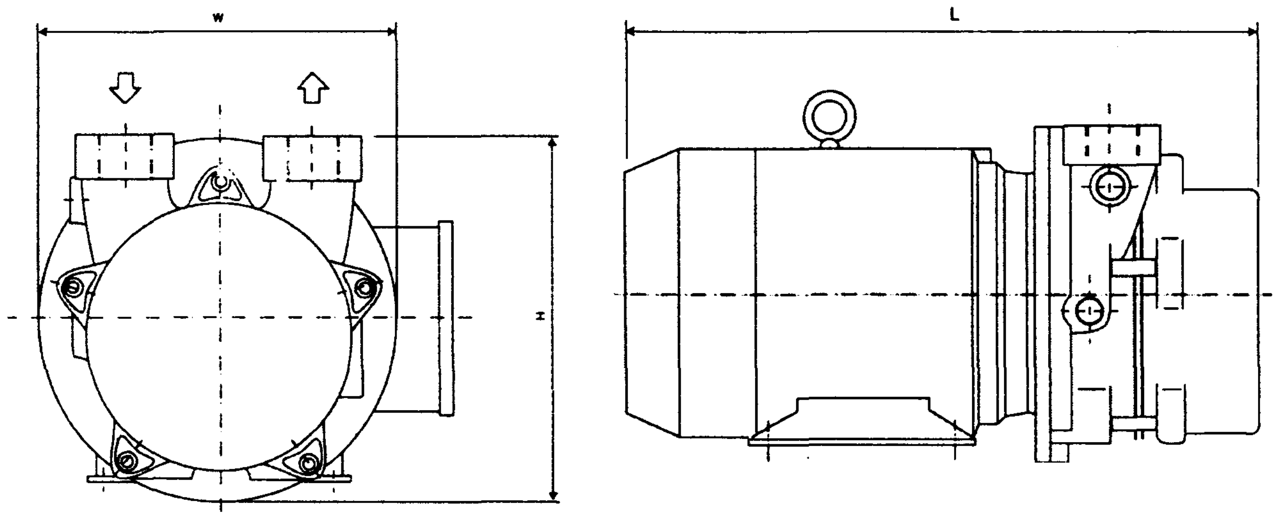
STOKES®
VACUUM Inc.
Stokes Vacuum Inc., 5500 Tabor Road, Philadelphia, PA 19120 USA
215-831-5400 Fax 215-831-5420 http://www.thomasregister.com/stokesvacuum

Performance Curves

Performance details of the CHR series single stage liquid ring pump with dry air at 20°C, service water at 15°C. 60 Hz performance.



Dimensions



Pump Size	CHR1015	CHR1027	CHR1055	CHR1100	CHR1200
Dimensions (in.) l x w x h	13 3/4 x 7 7/8 x 7 7/8	17 3/8 x 12 3/8 x 9 7/8	18 3/4 x 12 3/8 x 9 7/8	22 5/8 x 15 3/8 x 17 7/8	30 x 15 3/8 x 17 7/8
Suction & discharge connections (in.)	1 NPT	1 1/2 NPT	1 1/2 NPT	2 1/2 FLG	2 1/2 FLG
Total weight (lb)	51	99	110	279	381
Motor hp	2	5	7 1/2	7 1/2/10	10/20
Speed (rpm)	3400	3400	3400	1150/1740	1150/1740
Service water (gpm) Once through	1.8	6.0	6.0	9.0	13.7

STOKES® VACUUM Inc.

Stokes Vacuum Inc., 5500 Tabor Road, Philadelphia, PA 19120 USA
215-831-5400 Fax 215-831-5420 http://www.thomasregister.com/stokesvacuum

Appendix 3: PONDFEAS Information

Solar Pond Feasibility Study

location Bakersfield, California

Month	Average Pond Temp, C	Energy Required, GJ	Energy Extracted, GJ
Jan	74.3	0	0
Feb	73.9	0	0
Mar	77.3	0	0
Apr	78.5	272305	226297.6
May	80	272305	269932.9
Jun	83.8	272305	272305
Jul	87.4	272305	272305
Aug	88.9	272305	272305
Sep	87.1	272305	272305
Oct	81.5	272305	26877.1
Nov	79.5	0	0
Dec	77.1	0	0

Base Energy Required	1906136	
Solar Energy Supplied	1851228	
Purchased Energy	61008	
Annual Solar Fraction	0.97	
LCS with pond	236052144	
SIR	8.53	
Discounted Payback	1.3	
Levelized Cost of Energy	5.45	/GJ

Area	2500000	m^2
LCZ Depth	2	m
NCZ Depth	2	m
UCZ Depth	0.5	m
Excavation Vol	6951767	m^3
Pond Water Vol	(unreadable)	m^3
Wt of Salt	2042029	tonne
Liner Area	4096393	kg

Construction Costs ($)

excav cost	13903534
liner cost	36837536
salt cost	51050724
wave control	2500000
hx equip	7500000
control cost	2500
fence cost	167620
land cost	2072955
engineering	11406487
Total cost	125471360