THE PENNSYLVANIA STATE UNIVERSITY
SCHREYER HONORS COLLEGE

DEPARTMENT OF ENGINEERING SCIENCE AND MECHANICS

STEAM TURBINE PERFORMANCE MODELING VIA MATLAB SIMULATION PROGRAMMING WITH EMPHASIS ON LOW EXHAUST PRESSURE CONDITIONS

ZECHARIAH D. MEYER

Fall 2015

A thesis
submitted in partial fulfillment
of the requirements
for baccalaureate degrees
in Engineering Science and Energy Engineering
with honors in Engineering Science

Reviewed and approved* by the following:

Eric R. White
Senior Research Engineer
Thesis Supervisor

Gary L. Gray
Professor of Engineering Science and Mechanics
Honors Adviser

Judith A. Todd
P. B. Breneman Department Head Chair
Professor, Department of Engineering Science and Mechanics

* Signatures are on file in the Schreyer Honors College and the Engineering Science and Mechanics Office.
ABSTRACT

ARL has developed a turbine analysis FORTRAN program involving a closed-cycle Rankine system. It functions as a performance predictor by using turbine inlet pressure, turbine outlet pressure, turbine inlet temperature, turbine shaft rotational velocity, and geometric data to calculate torque, mass flow, and outlet temperature values. At low exhaust pressure conditions, the predictions for mass flow and torque are generally low compared to experimental data. To develop a modern, accurate turbine modeling tool usable for ARL in the future, the FORTRAN code was translated into MATLAB® and verified. A sensitivity analysis of the subroutines in the program was performed. Finally, simulation of experimental run data was performed and parameter/code modifications were tested to determine ways to improve the modeling system. A 4% increase in the input variable of turbine inlet pressure yielded accurate mass flow results and increased torque results. An increase of 0.0007” at the nozzle exit diameter and nozzle throat diameter parameters yielded accurate mass flow results and increased torque results from the program. Accounting for thermal expansion of the nozzle throat diameter yielded accurate mass flow results and increased torque results from the program. Recommendations were made to analyze the physical measurements used to determine inputs to the program, and to verify the validity of torque load assumptions that are used when actual torque cannot be directly measured.
TABLE OF CONTENTS

LIST OF FIGURES ........................................................................ iv
LIST OF TABLES ......................................................................... vii
ACKNOWLEDGEMENTS .............................................................. viii

Chapter 1 Introduction ................................................................ 1
  Problem Statement and Objectives ............................................. 2
  Literature Review ..................................................................... 3
  Design Needs .......................................................................... 8

Chapter 2 Methodology ............................................................... 9
  Translation of the Program from FORTRAN to MATLAB Code .... 9
  Verification of Results with MATLAB Code .............................. 11
  Sensitivity Analysis of Subroutines ......................................... 12
  Simulation of Experimental Data ............................................. 13
  Parameter Adjustment and Code Modification ......................... 14
  Recommendations for Improvement ....................................... 16

Chapter 3 Results .................................................................... 17
  Translation Verification Data ..................................................... 17
  Sensitivity Data ....................................................................... 18
  Experimental Data Comparisons ............................................ 39
  Parameter Adjustment and Code Modification Data ................ 44
  Analysis ................................................................................ 57
  Interpretation ......................................................................... 59

Chapter 4 Conclusions ............................................................... 60
  Summary ............................................................................... 60
  List of Technologically Important Findings .............................. 60
  Future Work Suggestions ....................................................... 61

Appendix A Team .................................................................... 62

Appendix B Tools ...................................................................... 63

Appendix C Broader Impacts ....................................................... 64
  Economic/Manufacturability/Technology Transfer .................... 64
  Environmental/ Sustainability ................................................ 64
  Ethical ................................................................................. 64
Health and Safety ................................................................. 65
Social/Political/Global .......................................................... 65

REFERENCES ........................................................................... 66
LIST OF FIGURES

Figure 1. Rankine Cycle Flow and Thermodynamic T-S Diagram. 2 .......................... 4
Figure 2. Turbine Wheel Blades. 7 ................................................................. 6
Figure 3. TRBTRQ Results: MATLAB vs. FORTRAN ................................. 18
Figure 4. ANOZP Sensitivity Analysis: PEXD vs. T ................................ 20
Figure 5. ANOZP Sensitivity Analysis: AMND vs. T ................................. 21
Figure 6. ANOZP Sensitivity Analysis: PEXD vs. P ................................ 22
Figure 7. ANOZP Sensitivity Analysis: AMND vs. P ................................ 23
Figure 8. TWOP Sensitivity Analysis: ETW0 vs. RHOG ............................ 24
Figure 9. TWOP Sensitivity Analysis: ETW0 vs. AMUG ............................ 25
Figure 10. TWOP Sensitivity Analysis: ETW0 vs. VSOND ......................... 26
Figure 11. TWOP Sensitivity Analysis: ETW0 vs. C ................................ 27
Figure 12. TWOP Sensitivity Analysis: ETW0 vs. RPM ............................ 28
Figure 13. ZNZPF Sensitivity Analysis: PHIN vs. PZOPE ......................... 29
Figure 14. ZNZPF Sensitivity Analysis: PHIN vs. PZOPED ....................... 30
Figure 15. CFSHST Sensitivity Analysis: PT vs. TIP ................................ 31
Figure 16. CFSHST Sensitivity Analysis: VELT vs. TIP ............................. 32
Figure 17. CFSHST Sensitivity Analysis: MDOAT vs. TIP ......................... 33
Figure 18. CFSHST Sensitivity Analysis: PT vs. TIT ................................ 34
Figure 19. CFSHST Sensitivity Analysis: VELT vs. TIT ............................. 35
Figure 20. CFSHST Sensitivity Analysis: MDOAT vs. TIT ......................... 36
Figure 21. VSONPS2 Sensitivity Analysis: VSONPS vs. S ......................... 37
Figure 22. VSONPS2 Sensitivity Analysis: VSONPS vs. P ......................... 38
Figure 23. Run A Mass Flow Program Results vs. Experimental Results ....... 40
Figure 24. Run A Torque Program Results vs. Experimental Results .......... 40
Figure 44. Run C Torque Program Results vs. Experimental Results with Nozzle Dimension Change. ..........................................................51

Figure 45. Run D Mass Flow Program Results vs. Experimental Results with Nozzle Dimension Change. ..........................................................52

Figure 46. Run D Torque Program Results vs. Experimental Results with Nozzle Dimension Change. ..........................................................52

Figure 47. Run A Mass Flow Program Results vs. Experimental Results with TIP Increase Modification. .............................................................53

Figure 48. Run A Torque Program Results vs. Experimental Results with TIP Increase Modification. .............................................................53

Figure 49. Run B Mass Flow Program Results vs. Experimental Results with TIP Increase Modification. .............................................................54

Figure 50. Run B Torque Program Results vs. Experimental Results with TIP Increase Modification. .............................................................54

Figure 51. Run C Mass Flow Program Results vs. Experimental Results with TIP Increase Modification. .............................................................55

Figure 52. Run C Torque Program Results vs. Experimental Results with TIP Increase Modification. .............................................................55

Figure 53. Run D Mass Flow Program Results vs. Experimental Results with TIP Increase Modification. .............................................................56

Figure 54. Run D Torque Program Results vs. Experimental Results with TIP Increase Modification. .............................................................56
LIST OF TABLES

Table 1. FORTRAN to MATLAB Verification: Input Parameter Ranges. .........................11
Table 2. Sensitivity Analysis Subroutine Information. ....................................................13
Table 3. Qualitative Comparisons of Predicted to Actual Mass Flow Data. ......................57
Table 4. Qualitative Comparisons of Predicted to Actual Torque Data. ...........................58
ACKNOWLEDGEMENTS

I would like to express my gratitude towards everyone who has aided me in completing this undergraduate thesis. Chiefly, I’d like to thank Eric White, my thesis supervisor. His guidance, expertise, and direction was essential to my success. Anytime I needed resources or assistance, he would either help me or point me in the right direction. Eric kept me on track and provided me with systematic goals to work towards. A very special thanks goes to Jeremy Walter, whose research and work on the turbine performance modeling program enabled my understanding of it and allowed me to make my own contributions. I must thank Ken Freeman for his assistance in analyzing my thesis data and providing insights into the computational significance of certain variable trends. I also appreciate Caroline Rice, my general supervisor, for seeking and finding opportunities for my thesis at Penn State’s Applied Research Laboratory (ARL). She helped me prioritize my work and set personal deadlines. Lastly, I would like to thank ARL as an organization for hiring me and provided me with the opportunities to do research for this undergraduate thesis.
Chapter 1

Introduction

I worked with the Integrated Power Systems Department of the Energy Science and Power Systems Division of the Advanced Technology Office of The Applied Research Laboratory (ARL) at The Pennsylvania State University. This research division focuses on research, development, and optimization of the power systems designed for underwater vehicles. Their work includes the modeling and simulation of various power cycles and their components. Specifically, I worked with a closed-cycle Rankine system and focused on analyzing and improving the turbine simulation models used as design tools and performance predictors.

The turbine analysis program I worked with takes four (4) input variables plus geometric data and outputs three (3) variables. The input variables are: 1) the turbine inlet pressure, 2) inlet temperature, 3) outlet pressure, and 4) revolutions per minute (RPM). The output variables are: 1) torque, 2) mass flow, and 3) outlet temperature. Of these outputs, torque and mass flow are the most important because they can actually be measured physically. This program is used iteratively to determine RPM from experimental turbine inlet pressure, turbine inlet temperature, turbine exhaust pressure data, and assumed torque values.
Problem Statement and Objectives

The primary research problem statement was twofold:

1. What existing program assumptions are limiting the accuracy of the predictions at low turbine exhaust pressures?
2. What can be done to improve upon these issues in the new MATLAB code?

The objectives that were to be completed in answering the problem statement included:

1. Translation of the program from FORTRAN to MATLAB code
2. Verification of the results from the new program code
3. Sensitivity analysis of the subroutines within the program
4. Simulation of actual experimental data with the new program
5. Parameter adjustment and code modification analysis
6. Recommendations for improvement in the performance modeling program

The rationale for conducting this work was to create improved models so that more accurate performance predictions could be made regarding the Rankine system of which the turbine is a component.
Literature Review

Turbines convert energy from fluid flow into rotational mechanical energy. Steam turbines are a specific type of turbine that use superheated water vapor as the working fluid. The Rankine cycle is the most common thermodynamic cycle for heat engines that employ steam turbines to convert fluid energy into mechanical energy. The performance modeling program referenced in this thesis deals with a turbine found within a Rankine system.

The major components in a Rankine Cycle include a steam generator (boiler), a turbine, a condenser, and a pump. This is a closed loop through which water flows and transitions between liquid and gas states. As shown below in Figure 1, the superheated steam enters the turbine at state 1. Ideally, this superheated steam isentropically expands (meaning without a change in entropy) through the turbine, losing temperature and pressure, until it exits the turbine to reach state 2. The process from state 1 to state 2 is where mechanical work is produced. In many applications other than the application of this thesis topic, such energy is then used to generate electricity. From state 2 to state 3, the working fluid transitions from gas to liquid state in the condenser, staying at constant temperature and pressure but rejecting heat and losing entropy. At this point, the liquid water is pushed through the pump, where it is raised to a high pressure and slightly increases in temperature, moving from state 3 to state 4. Finally, to complete the cycle, the liquid goes through the boiler where it receives heat, is vaporized, increases in temperature and entropy, and transitions to the superheated state 1 at the turbine’s inlet.
Figure 1. Rankine Cycle Flow and Thermodynamic T-S Diagram.

It should be noted that the Rankine cycle shown in Figure 1 is actually an enhanced Rankine cycle because it utilizes the common improvement of superheating the steam involved. The Rankine cycle of the turbine analyzed in this research utilizes this improvement as well. Superheated steam is defined as steam heated above saturation temperature—the temperature at which any decrease in temperature at constant pressure will result in condensation of the steam. Below the saturation temperature, steam is two-phase (meaning partially liquid and partially vapor), and it has a property called quality, which is defined as the mass fraction of steam which
is vapor. A quality of one means 100% saturated vapor, while a quality of zero means 100% saturated liquid. Superheating the steam accomplishes two major things. First, it increases the mean temperature of the steam when it is heated, thus improving the thermodynamic efficiency of the cycle. Second, it increases the quality of the steam when it moves through the turbine blades. This increase in quality means a decrease or possible elimination of liquid in the spinning turbine blades, which should reduce the deterioration of the blades and prolong their life. The presence of superheated steam in the turbine is important to take into account for reasons that will be described later in this thesis.

There are standard thermodynamic steam tables that generally must be referenced to calculate the specific changes in thermodynamic properties like internal energy, enthalpy, entropy, volume, etc. of steam as it changes in temperature, pressure, and state. Alternatively, complex equations of state can be used to arrive at the steam properties found in such tables. Since the program analyzed here uses many iterative calls to retrieve this thermodynamic data, it made most sense to use a designated thermodynamic function program for these calculations. An open-source MATLAB program called XSteamUS.m (the US designation for imperial units) works well for this purpose and was incorporated into the program. The XSteamUS.m program uses steam and water properties from the “International Association for Properties of Water and Steam Industrial Formulation 1997” (IAPWS-97). For the purpose of this research, XSteamUS.m generally input pressure and/or temperature, pressure and enthalpy, or pressure and entropy to produce any other desired thermodynamic property.

In reality, to accurately analyze the thermodynamics of the Rankine cycle, additional analyses must be done on each component; the component in question was the turbine for this thesis research. As already established, a turbine functions by converting fluid flow energy into
mechanical energy. Specifically, pressurized, heated steam enters the turbine nozzles to be accelerated to high velocity and then passes through the turbine wheel blades, which dictate the path through which the steam travels.\(^1\) An example of turbine blades can be seen in Figure 2.\(^7\)

![Figure 2. Turbine Wheel Blades.\(^7\)](image)

As the steam pushes against the blades, it induces a torque which then turns the shaft about which the turbine blades are radially placed.\(^1\) Prior to entering the turbine blades, the steam passes through a choked nozzle. Naturally, to understand the flow of this fluid, fluid mechanics must be studied and utilized. Along such a path in the turbine, the water vapor/steam goes through changes in density because this working fluid is a gas, and gas is compressible.\(^8\) Thus, a large portion of the MATLAB program code is dedicated to compressible fluid flow.

The calculations done with compressible fluid flow in many of the nozzle flow subroutines within the program consider six fundamental assumptions. The first assumption is that the gas is continuous.\(^9\) This means that the motion of single molecules does not need to be considered, and thus inherently assumes that the gas is not at very low pressure and density. Single gas molecule motion need not be accounted for when the mean free path of the gas
molecules is insignificant compared to the dimensions of the mechanical part through which it flows. The second assumption is that no chemical changes occur in the gas. This is a non-issue because the Rankine cycle is a closed-loop system with only physical phase changes happening to the water/steam. The third assumption for some of the calculations of supersonic flow is that the gas is perfect and follows the ideal gas law. Although the steam in the system is not a perfect gas, more detailed calculations in the supersonic nozzle would severely complicate compressible flow equations with a very limited improvement in modeling accuracy. The fourth assumption is that gravity effects in the gas flow are negligible, which is a justified assumption based on the order of magnitude difference between gravitational and gas pressure forces in the turbine. The fifth assumption is that electromagnetic effects are negligible – again justified because the steam involved is not ionized or directly influenced by an electromagnetic field. The final assumption is that the effects of viscosity are negligible, which is the only assumption that can be relaxed in some cases. However, viscosity is most relevant with near-solid materials and generally has very small effects on gases.

Another assumption for the aforementioned nozzle is that of isentropic flow, meaning flow with a constant state of entropy. This assumption has been validated by performance observations of well-designed supersonic steam nozzles. Since the nozzle in the turbine fits this description, the assumption is valid.

Finally, in the research I conducted, thermal expansion was considered as an effect on the turbine calculations. As materials increase in temperature, their size increases proportionately by a factor known as the coefficient of expansion. This consideration assumed linear expansion, which can be described by the in equation 1.
\[ L = L_0 (1 + \alpha \Delta T) \]  \hspace{1cm} (1)

where \( L \) is the final length, \( L_0 \) is the initial length, \( \alpha \) is the coefficient of linear expansion, and \( \Delta T \) is the change in temperature.

**Design Needs**

The turbine performance model analyzed has slight accuracy issues with predictions at low turbine exhaust pressures. This thesis research was in analyzing the first principle and empirical formulas as well as parameter assumptions used in the program and comparing their accuracy with experimental data to find improvements to the modeling technique. In addition, the program was translated from an older coding language (FORTRAN) to a more modern one (MATLAB). Effectively, this research created an improved version of the original turbine performance modeling tool.
Chapter 2
Methodology

The following sections describe the details of each of the steps that were followed during the completion of this research.

1. Translation of the program from FORTRAN to MATLAB code
2. Verification of the results from the new program code
3. Sensitivity analysis of the subroutines within the program
4. Simulation of actual experimental data with the new program
5. Parameter adjustment and code modification analysis
6. Recommendations for improvement in the performance modeling program

Translation of the Program from FORTRAN to MATLAB Code

This step involved translating over 1,000 lines of code of a parent program and 23 subroutines from the outdated FORTRAN language into an updated MATLAB language. Translation allowed for the program to be more easily incorporated into separate MATLAB analysis scripts. This step generally entailed using the same variable names, but referencing and using them as MATLAB requires. In MATLAB, unlike FORTRAN, variable precisions do not need to be explicit every time a variable is referenced. However, MATLAB code generally requires a semicolon at the end of each line unlike FORTRAN. The function names for
operators such as square root are different for both languages, and loop structures are handled entirely differently. MATLAB has the ease of using a “for” loop, where a section of code will repeat until a specified condition is met. Visually, this allows for nesting of functions and keeps the code graphically organized. The original FORTRAN code, however, uses conditional branching to statement labels. Statement labels work by giving a certain line of code a number label and allowing other lines of code to jump to those lines with the “GOTO” operator followed by referencing that number label. Translating such commands requires step-by-step tracking of the code to map out each combination of how the code would read based on which conditions were met and in what order. Logic structures were created in MATLAB to match each of the FORTRAN loop structures.

Another major change from the FORTRAN to the MATLAB included implementing XSteamUS.m for thermodynamic function calls instead of attempting to re-code all thermodynamic calculations in the FORTRAN steam subroutine library. However, a few subroutines needed to be translated by hand. These included the calculations for sonic velocity and critical flow parameters of superheated steam through a choked nozzle.

Lastly, the code was broken up from one “.f” file in FORTRAN to individual functions for all subroutines and the parent program in MATLAB. The FORTRAN version used a text file and a compiler to run the program iteratively, whereas the new MATLAB version was coded to use a master script “.m” file. The master script iteratively calls the parent program function that subsequently calls any of the other subroutine functions. This allowed for a much cleaner and easily customizable (via modification to the master “.m” script) user interface.
Verification of Results with MATLAB Code

Upon finishing the translation, the MATLAB code had to be compared to the legacy FORTRAN code using realistic input ranges. Ideally the results should have been identical, but the average percent difference in results was 0.17%; this was deemed an acceptable variance in results. Comparison was done by setting the loop structure in the master MATLAB script to run identically to the loop structure used by the FORTRAN compiler version, then collecting the results in a Microsoft Excel spreadsheet and determining the percent difference in results based on the original FORTRAN results. The cases were based on the input parameters shown in Table 1 below.

<table>
<thead>
<tr>
<th>Input Parameter</th>
<th>Range</th>
<th>Increments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inlet Pressure (psia)</td>
<td>200-800</td>
<td>300</td>
</tr>
<tr>
<td>Inlet Temperature (°F)</td>
<td>1,400-1,600</td>
<td>100</td>
</tr>
<tr>
<td>Outlet Pressure (psia)</td>
<td>3-17</td>
<td>7</td>
</tr>
<tr>
<td>Turbine Rotational Speed (RPM)</td>
<td>84,000-132,000</td>
<td>24,000</td>
</tr>
</tbody>
</table>

The mass flow and torque results were graphed against arbitrary case numbers of the different input parameter combinations from both the MATLAB and the FORTRAN results to graphically show that the data matched; this will be presented in Chapter 3.
Sensitivity Analysis of Subroutines

Upon verifying the results from the new MATLAB version of the program, the major individual subroutines were analyzed to determine the effect each input parameter had on the relevant output parameters. This was done by varying one input parameter at a time while holding the other input parameters constant at an average input value, then observing how and by how much the output variables change. Multiple scripts were coded for this purpose. This analysis was performed on five subroutines: three subroutines from the turbine analysis program encompassed all the other minor subroutines, and the other two were thermodynamic function calls to supplement XSteamUS. The three subroutines from the turbine analysis program were named: TWOP, which calculates the turbine wheel operating performance; ANOZP, which calculates the nozzle design exit pressure; and ZNZPF, which estimates the real nozzle performance. The two thermodynamic function calls were: CFSHST, which calculates the critical flow parameters of superheated steam through a choked nozzle; and VSONPS2, which calculates sonic velocity. The subroutine names which are somewhat non-descriptive were held over from the FORTRAN code which had name length restrictions. Table 2 below shows the input variables and output variables for each of the 5 subroutines.
Table 2. Sensitivity Analysis Subroutine Information.

<table>
<thead>
<tr>
<th>Subroutine</th>
<th>Input Variables</th>
<th>Output Variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANOZP</td>
<td>P, T, AOAST</td>
<td>PEXD, AMND</td>
</tr>
<tr>
<td>TWOP</td>
<td>RHOG, AMUG, VSOND, C, RPM</td>
<td>ETW0</td>
</tr>
<tr>
<td>ZNZPF</td>
<td>G, PZOPED, PZOPE, PHIND</td>
<td>PHIN</td>
</tr>
<tr>
<td>CFSHST</td>
<td>P, T</td>
<td>PT, VELT, MDOAT</td>
</tr>
<tr>
<td>VSONPS2</td>
<td>P, S</td>
<td>VSONPS</td>
</tr>
</tbody>
</table>

All input variables ranges for the sensitivity analysis of these subroutines were selected to encompass or represent the full range of standard experimental testing conditions. These ranges were found by tracking the effects of the parent program input condition ranges (i.e. inlet pressure, inlet temperature, RPM, and outlet pressure) down through the code to determine the ranges of inputs to each subroutine.

Simulation of Experimental Data

Experimental data from four previously conducted turbine tests (designated A, B, C, and D) were collected, organized, and imported into MATLAB structures. These data included the four input parameters (and other data) collected at a time interval 0.05 seconds. A master script was created that would read and collect these data from the MATLAB structures. It would run the analysis program at each time interval and record the time, input parameters, simulated output variables, and measured output variables from the experimental testing. Thus the
measured output variables for torque and mass flow were the expected results, and the program results for mass flow and torque could be compared graphically to the experimental results over the duration of the test. However, since the new thermodynamic function calls used XSteamUS instead of the older code, some calculations sent the analysis program into an endless error loop. The problem was that the previous thermodynamic code could extrapolate for superheated steam calls when the data points were really in the saturated region. The new XSteamUS calls, although more accurate, could not do so. This mandated that the analyzed time ranges of the experimental data had to be selected to not include some of the data near the beginning and end of the tests when the turbine was not in steady state operation.

**Parameter Adjustment and Code Modification**

After analyzing the first principle and empirical assumptions present in the parent program, the method seemed to be sound and no obvious deficiencies were found in the compressible flow calculations. Instead, two possible alternatives to direct code modification were investigated to improve the accuracy of the results. One was to examine the effect that various parameters had on the results, and the other was a code addition to account for the thermal expansion of the nozzle at the high inlet temperatures experienced due to the superheated steam.

For the first case, the various parameters that were tested included variations of the turbine inlet pressure, turbine inlet temperature, the dimensions of the nozzle (DNT and DNE), the blade roughness (BRUF), the nozzle mass flow coefficient (CMFC), and the on-design nozzle velocity coefficient (PSIND). Each of these parameters was isolated and evaluated via a
guess-and-check method, manually changing the parameters and observing the resulting overlaid graphs, then modifying the parameter again and re-testing to more closely match the experimental data. This process was manual and not automated due to the short program runtime and the ease in which these parameters could be changed. Most parameters were variables initiated at the beginning of the parent function so they could not be iteratively changed in the master script. Each parameter was isolated and modified such that the resulting mass flow and torque graphs would best match, with preference given to the mass flow results because the experimental data regarding mass flow were direct, while the experimental data for torque were inferred from the measured data and assumed load characteristics.

For the second case, the linear expansion modification assumed linear thermal expansion, as described in the previous literature review section. The length was actually the nozzle throat diameter (DNT). The coefficient of thermal expansion assumed a factor previously determined at ARL. Finally, the change in temperature was the difference between an average room temperature of 70°F and the turbine inlet temperature. This thermal expansion addition to the code was added immediately after the initiation of the DNT variable and was updated throughout the simulation as the turbine inlet temperature changed. The simulation results were compared to the base case for each set of test data.

Numerical analysis of the resulting graphs was not necessary, since the results were based on hypothetical parameter changes that were manually estimated to yield the best results. They were not physically tested parameter changes, and their determination was by iterative means. Thus qualitative analysis was appropriate to analyze the effect of each parameter change or thermal expansion code addition on the resulting graphs.
Recommendations for Improvement

After noting the effects of thermal expansion and the variations of parameters, research into the physical significance of the potential improvements was done via inspection of the physical turbine system. The conclusions made from this allowed for recommendations for improvement. However, since the problem statement for this thesis was explorative in nature, the results were not conclusive in identifying a single problem area. Rather, a few suggestions were made for future investigation.
Chapter 3

Results

Translation Verification Data

Figure 3 demonstrates a visual representation of the matching results from the MATLAB and FORTRAN versions of the program. The case numbers on the x-axis are arbitrary combinations of the input parameters found in Table 1, and the resulting mass flow and torque data points from the MATLAB code overlap the FORTRAN data points. This shows that the code produces the same results with both languages.
Figure 3. TRBTRQ Results: MATLAB vs. FORTRAN.

Sensitivity Data

The five key subroutines of the program that were analyzed were ANOZP, TWOP, ZNZPF, CFHSST, and VSONPS2, as described in the sensitivity analysis section of Chapter 2. ANOZP inputs the stagnation pressure at the nozzle inlet ($P_0$), the stagnation temperature at the nozzle inlet ($T_0$), and the exit/throat area ratio (AOAST). It uses real steam properties to calculate the isentropic pressure at the exit plane of the nozzle (PEXD) and the isentropic Mach number at the exit plane of the nozzle (AMND). The relevant inputs to TWOP are the density of
steam at the nozzle exit (RHOG), viscosity of steam at the nozzle exit (AMUG), velocity of sound at the nozzle exit (VSOND), nozzle exit velocity (C), and revolutions per minute (RPM), with the key output being a baseline value for turbine wheel efficiency (ETW0). ZNZPF inputs the ratio of specific heats (G) for superheated steam, the designed nozzle pressure ratio (PZOPED), the actual nozzle pressure ratio (PZOPE), and the on-design nozzle velocity coefficient (PHIND). It outputs the adjusted nozzle velocity coefficient at current conditions (PHIN). CFSHST is a function that computes the critical flow parameters for superheated steam through a choked nozzle using the thermodynamic properties of steam. The inputs are inlet stagnation pressure (P0) and temperature (T0). The outputs are the pressure at the throat (PT), the velocity at the throat (VELT), and the mass flow rate per unit area at the throat (MDOAT). VSONPS2 computes the sonic velocity for steam at a given pressure (P) and entropy (S). The following figures and explanations show how each input parameter to these subroutines affects each output variable.
As seen in Figure 4, the nozzle design exit pressure (PEXD) dependence on inlet temperature (T) appeared fairly linear for pressures in the range of 225 to 925 psia, but a 2nd order polynomial trend line fit the data better. At higher pressures (P), the PEXD variable did not behave as simply. The trend line for an inlet pressure of 1225 psia was best modeled by a 6th order polynomial as noted on the graph. PEXD ranged from ~3 to 18 psia when reasonable pressures were used. PEXD increased slightly with increases in inlet temperature, but it increased sharply with increases in inlet pressure.

ANOZP is driven purely by physics – there is nothing to adjust inside this subroutine other than computation accuracy of the steam properties.
Figure 5 demonstrates the exit Mach number (AMND) dependence on inlet temperature (T), which is more complex than that of the nozzle design exit pressure (PEXD) dependence. The best fit trend lines used 6th order polynomial functions. AMND had a range of approximately 3.25 to 4.5 when reasonable pressures (P) were used. Most AMND values ranged from approximately 3.25 to 3.75 when pressures higher than 225 psia were used. AMND decreased slightly with increases in temperature, and it decreased slightly with increases in pressure as well.
The nozzle design exit pressure (PEXD) values were linearly and proportionately related to pressure (P), with a strong $R^2 = 1$ correlation, as shown in Figure 6. As P increased, PEXD increased with a slope of 0.0145. PEXD was minimally affected by temperature.

If steam were a perfect gas, the isentropic pressure ratio is determined exactly by the area ratio, meaning that all these would fall on the same line for the same value of gamma. The real properties of steam cause the small variation.

**Figure 6.** ANOZP Sensitivity Analysis: PEXD vs. P.
Figure 7. ANOZP Sensitivity Analysis: AMND vs. P.

Figure 7 shows that exit Mach number (AMND) was dependent upon pressure by a 6th order polynomial function. Temperature had little effect on AMND. At low pressures, small changes in pressure yielded large decreases in AMND. At pressures greater than approximately 400psia, AMND stayed fairly constant at around 3.3.
Figure 8 demonstrates the 6th order polynomial relationship between turbine wheel efficiency (ETW0) and the density of steam at the nozzle exit (RHOG). The smaller the value of RHOG was, the larger effect a change in RHOG had on ETW0. ETW0 increased at a decreasing rate with increases in RHOG. It should also be noted that since ETW0 is an efficiency, variations in RHOG affected ETW0 by less than 0.5%. The legend in Figure 8 describes the conditions of the other input parameters to TWOP while the effect of RHOG on ETW0 was isolated. They are all at values close to their median when the turbine is operated at typical conditions.
Turbine wheel efficiency (ETW0) was linearly related to the viscosity of steam at the real nozzle exit (AMUG), as shown in Figure 9. ETW0 decreased with increases in AMUG, with a slope of -4665.2. However, much like the case with RHOG, ETW0 was not significantly affected by AMUG. ETW0 changed less than 0.1% due to reasonable changes in AMUG. The legend in Figure 9 describes the conditions of the other input parameters to TWOP while the effect of AMUG on ETW0 was isolated. They are all at values close to their median when the turbine is operated at typical conditions.

Figure 9. TWOP Sensitivity Analysis: ETW0 vs. AMUG.
Turbine wheel efficiency (ETW0) was related to the velocity of sound in steam at the real nozzle exit (VSOND) via a 6th order polynomial trend line. ETW0 increased at a decreasing rate as VSOND increased, and ETW0 plateaued at an apparent maximum around 0.788. The range of ETW0 with reasonable VSOND values was between about 0.755 and 0.788 when the other input variables were held to the values found in the legend of Figure 10. ETWO changed by less than 4% when VSOND was varied between 1250 and 2600 ft/s.

Figure 10. TWOP Sensitivity Analysis: ETW0 vs. VSOND.
Of all of TWOP’s input parameters, turbine wheel efficiency (ETW0) was affected most by nozzle exit velocity (C). With the other input parameters held at reasonable, median values, ETW0 varied from 0.6 to 0.8. This was a difference of about 20% efficiency. The ETW0-C relationship was also complex in shape: as shown in Figure 11, a 6th degree polynomial function was needed to model the data. ETW0 peaked at 80% when C had a value of around 3500ft/s. Having a C value of less than or greater than that peak resulted in lower ETW0 values. The ETW0 value dropped off more steeply when C was less than its peak value.

Figure 11. TWOP Sensitivity Analysis: ETW0 vs. C.
Revolutions per minute (RPM) of the turbine also had a large effect on turbine wheel efficiency (ETW0), as shown in Figure 12. Best modeled by a 6th degree polynomial, ETW0 peaked at 80% with an RPM of 141600. With an RPM less than that or greater than that, ETW0 decreased at an increasing rate. ETW0 ranged from 65% to 80% as RPM varied from approximately 84,000 to 168,000.

Figure 12. TWOP Sensitivity Analysis: ETW0 vs. RPM.
Nozzle velocity coefficient (PHIN) was related to the actual inlet/outlet pressure ratio (PZOPE) via a 6th degree polynomial function. As PZOPE increased, PHIN increased at a decreasing rate, then slowly decreased after PZOPE ≈ 75. Figure 13 shows that PHIN varied from 0.6 to 1 as PZOPE varied from 0 to 425 and the design inlet/outlet pressure ratio (PZOPED) remained constant at 70.

**Figure 13.** ZNZPF Sensitivity Analysis: PHIN vs. PZOPE.
Figure 14. ZNZPF Sensitivity Analysis: PHIN vs. PZOPED.

Figure 14 shows that the nozzle velocity coefficient (PHIN) appeared to be related to the design inlet/outlet pressure ratio (PZOPED) linearly, but in fact, a quadratic function fits the data better. PHIN increased from approximately 0.95877 to 0.95925 as PZOPED increased from about 68.5 to 72.5. As such, PHIN hardly changed from the effects of PZOPED. Here, the actual inlet/outlet pressure ratio (PZOPE) held constant at 87.5.
There was a linear relationship between the pressure at the nozzle throat (PT) and the inlet pressure (TIP) in CFSHST. PT increased at a slope of 0.5537 compared to TIP when the inlet temperature (TIT) was held at 1550 °F. PT ranged from 100 to 700 psi as TIP ranged from 200 to 1300 psi. Thus, as shown in Figure 15, PT is heavily and predictably reliant on TIP.

**Figure 15.** CFSHST Sensitivity Analysis: PT vs. TIP.
Figure 16 shows that there was a linear relationship between the velocity of flow in the nozzle throat (VELT) and turbine inlet pressure (TIP) in CFSHST. VELT decreased at a slope of -0.014 compared to TIP when turbine inlet temperature (TIT) was held at 1550°F. VELT ranged from 2486 to 2470 ft/s as TIP ranged from about 200 to 1300 psi. Thus, VELT was not heavily reliant upon TIP.

Figure 16. CFSHST Sensitivity Analysis: VELT vs. TIP.
There was a linear relationship between the mass flow per area through the nozzle (MDOAT) and turbine inlet pressure (TIP) in CFSHST. MDOAT increased at a slope of 0.0091 compared to TIP when turbine inlet temperature (TIT) was held at 1550°F. MDOAT ranged from less than 2 to slightly less than 12 lbm/s/ft² as TIP ranged from about 200 to 1300. Although the range of MDOAT was small, the values were small as well. Figure 17 demonstrates that MDOAT was heavily reliant upon TIP.

Figure 17. CFSHST Sensitivity Analysis: MDOAT vs. TIP.
Figure 18 shows that the throat pressure (PT) is independent of turbine inlet temperature (TIT) when turbine inlet pressure (TIP) is held at 725psia. The value of PT here was 401.43psi.
Although almost linear, velocity in the nozzle throat (VELT) was related to turbine inlet temperature (TIT) in a quadratic fashion. Figure 19 shows that as TIT increased from 1300°F to 1800°F, VELT increased from approximately 2300 to 2650 ft/s. Thus, VELT was strongly dependent upon TIT.
Figure 20. CFSHST Sensitivity Analysis: MDOAT vs. TIT.

Figure 20 demonstrates the inverse relationship between the mass flow per area (MDOAT) and turbine inlet temperature (TIT). As TIT increased from 1300°F to 1800°F, MDOAT decreased from about 7 to less than 6.2 lbm/s/ft². MDOAT was moderately dependent upon TIT.
Figure 21. VSONPS2 Sensitivity Analysis: VSONPS vs. S.

As seen in Figure 21, sonic velocity (VSONPS2) increased as entropy (S) increased. However, the way it did so was subject to the effects of pressure. When pressure was lower (i.e. at $P=2.5\text{psia}$), the relationship between VSONPS2 and S was modeled best by a 4th order polynomial equation. As pressure ($P$) increased to 14.5$\text{psia}$, this relationship was modeled better by a linear function, and all VSONPS2 values were shifted to be larger. Thus, with increasing pressure, the relationship between VSONPS and S became increasingly linear. Also, VSONPS2 was highly dependent upon S. Each increase of 0.5 in S resulted in at least an increase of 500 in VSONPS2, regardless of pressure.
As pressure (P) increased, sonic velocity (VSONPS2) increased at a decreasing rate. Figure 22 shows this as best being modeled by a 6th order polynomial function. Higher values of S shifted this function to higher VSONPS2 levels. An increase of about 15psia in P resulted in an increase of at least 500ft/s in VSONPS2, so VSONPS2 was highly dependent on P.
Experimental Data Comparisons

The following four pages include the baseline graphs of how the program predicted mass flow through the turbine nozzles and generated torque compared to the expected results for runs A, B, C, and D. Each page is dedicated to a run and includes both the mass flow and torque figures. In every case, mass flow is consistently predicted low. Torque is generally predicted low as well, but on runs B and D, it predicts accurately for at least part of the test.
**Figure 23.** Run A Mass Flow Program Results vs. Experimental Results.

**Figure 24.** Run A Torque Program Results vs. Experimental Results.
Figure 25. Run B Mass Flow Program Results vs. Experimental Results.

Figure 26. Run B Torque Program Results vs. Experimental Results.
Figure 27. Run C Mass Flow Program Results vs. Experimental Results.

Figure 28. Run C Torque Program Results vs. Experimental Results.
Figure 29. Run D Mass Flow Program Results vs. Experimental Results.

Figure 30. Run D Torque Program Results vs. Experimental Results.
Parameter Adjustment and Code Modification Data

The conditions that were varied in simulation included the turbine inlet pressure (TIP), turbine inlet temperature (TIT), the dimensions of the nozzle (DNT and DNE), blade roughness (BRUF), the nozzle mass flow coefficient (CMFC), and the nozzle velocity coefficient on design (PSIND). The code modification for thermal expansion of the throat diameter was tested as well. From this list, only 3 modifications were viable options to help correct the program’s predictions. These were the addition of the thermal expansion code, modification of the nozzle dimensions (DNT and DNE), and modification of the inlet pressure (TIP). Modifications to TIT by means of up to a ±20% only decreased the accuracy of the program’s predictions. Changing the blade roughness factor by orders of magnitude only presented negligible effects on the program’s predictions for mass flow or torque. Likewise, doubling or halving the PSIND factor of 0.96 produced no noticeable changes. Increasing the CMFC coefficient from 0.98 to 1 produced a minimal accuracy increase for mass flow predictions, but it is an unrealistic change because a CMFC coefficient of 1 assumes ideal behavior.

The modifications that yielded the most accurate mass flow results included increasing TIP input values by 4%, adding the correction for thermal expansion at the nozzle throat diameter (DNT), or adding 0.0007” to the diameter of both the nozzle throat and exit diameters (to effectively widen the nozzle without proportionately scaling the nozzle shape). The thermal expansion modification only addressed DNT because this dimension is far more important for the compressible flow calculations than DNE, and it saves computation time to only address one dimension. The following 12 pages include graphs to show the results of each of these modifications on mass flow and torque for Runs A, B, C, and D.
Figure 31. Run A Mass Flow Program Results vs. Experimental Results with Thermal Expansion Code Addition.

Figure 32. Run A Torque Program Results vs. Experimental Results with Thermal Expansion Code Addition.
Figure 33. Run B Mass Flow Program Results vs. Experimental Results with Thermal Expansion Code Addition.

Figure 34. Run B Torque Program Results vs. Experimental Results with Thermal Expansion Code Addition.
**Figure 35.** Run C Mass Flow Program Results vs. Experimental Results with Thermal Expansion Code Addition.

**Figure 36.** Run C Torque Program Results vs. Experimental Results with Thermal Expansion Code Addition.
**Figure 37.** Run D Mass Flow Program Results vs. Experimental Results with Thermal Expansion Code Addition.

**Figure 38.** Run D Torque Program Results vs. Experimental Results with Thermal Expansion Code Addition.
Figure 39. Run A Mass Flow Program Results vs. Experimental Results with Nozzle Dimension Change.

Figure 40. Run A Torque Program Results vs. Experimental Results with Nozzle Dimension Change.
Figure 41. Run B Mass Flow Program Results vs. Experimental Results with Nozzle Dimension Change.

Figure 42. Run B Torque Program Results vs. Experimental Results with Nozzle Dimension Change.
Figure 43. Run C Mass Flow Program Results vs. Experimental Results with Nozzle Dimension Change.

Figure 44. Run C Torque Program Results vs. Experimental Results with Nozzle Dimension Change.
Figure 45. Run D Mass Flow Program Results vs. Experimental Results with Nozzle Dimension Change.

Figure 46. Run D Torque Program Results vs. Experimental Results with Nozzle Dimension Change.
Figure 47. Run A Mass Flow Program Results vs. Experimental Results with TIP Increase Modification.

Figure 48. Run A Torque Program Results vs. Experimental Results with TIP Increase Modification.
Figure 49. Run B Mass Flow Program Results vs. Experimental Results with TIP Increase Modification.

Figure 50. Run B Torque Program Results vs. Experimental Results with TIP Increase Modification.
**Figure 51.** Run C Mass Flow Program Results vs. Experimental Results with TIP Increase Modification.

**Figure 52.** Run C Torque Program Results vs. Experimental Results with TIP Increase Modification.
**Figure 53.** Run D Mass Flow Program Results vs. Experimental Results with TIP Increase Modification.

**Figure 54.** Run D Torque Program Results vs. Experimental Results with TIP Increase Modification.
Analysis

Since there was no specified accuracy goal for the program results, the results could be analyzed via a simple visual analysis of the resulting graphs. A percent-difference analysis of the results, while quantifying the data, would rely on the modifications being exact in nature to have practical significance. Since the parameter changes were determined (out of necessity) via iterative guess-and-check and were explorative in nature, this quantitative analysis was not necessary.

Table 3 demonstrates the effects of all three effective parameter changes on mass flow results. The notes in the table are relations of the program predictions to the experimental results for each run. For example, simulating Run A with the unmodified program results in program predictions that were low compared to the experimental data throughout the run. Each of the three parameter changes increased the program’s results to better match the experimental data for mass flow.

<table>
<thead>
<tr>
<th>Run</th>
<th>Unmodified Program</th>
<th>4% TIP Increase</th>
<th>Thermal Expansion</th>
<th>DNE and DNT increase, 0.0007”</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Low</td>
<td>Acceptable Match</td>
<td>Acceptable Match</td>
<td>Acceptable Match</td>
</tr>
<tr>
<td>B</td>
<td>Low</td>
<td>Acceptable Match</td>
<td>Acceptable Match</td>
<td>Acceptable Match</td>
</tr>
<tr>
<td>C</td>
<td>Low</td>
<td>Acceptable Match</td>
<td>Acceptable Match</td>
<td>Acceptable Match</td>
</tr>
<tr>
<td>D</td>
<td>Low</td>
<td>Acceptable Match</td>
<td>Acceptable Match</td>
<td>Acceptable Match</td>
</tr>
</tbody>
</table>
The effect of the parameter changes on torque provided less effective result improvements but better insight into the program. It should be noted that all parameter modifications increased the torque results throughout each run. For runs A, B, and C, this was a step closer to matching the experimental results in which torque was not directly measured, but inferred from measured speed. As shown in Table 4, the only situation where a parameter modification provided an adequate match to the experimental results was for time > ~60 seconds on Run B, with the 4% TIP increase. However, the unedited program actually predicted acceptable torque values for Run D, in which torque could be directly measured. Since the tested parameter changes all increased the torque results for every case, they also effectively produced less accurate torque results compared to the unedited program in Run D.

<table>
<thead>
<tr>
<th>Run</th>
<th>Unmodified Program</th>
<th>4% TIP Increase</th>
<th>Thermal Expansion</th>
<th>DNE and DNT increase, 0.0007”</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td>B</td>
<td>High then Low</td>
<td>High then Acceptable Match</td>
<td>High then Low</td>
<td>High then Low</td>
</tr>
<tr>
<td>C</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td>D</td>
<td>Acceptable Match</td>
<td>High</td>
<td>High</td>
<td>High</td>
</tr>
</tbody>
</table>

Table 4. Qualitative Comparisons of Predicted to Actual Torque Data.
Interpretation

Accounting for thermal expansion, increasing the nozzle throat and exit diameters, and increasing turbine inlet pressure all improve the mass flow prediction accuracy and increase the torque results. The increase in torque results was beneficial for runs A, B, and C, but detrimental for Run D. Assuming that at least one of the tested parameter changes was a valid change to the program, there exists a possibility that the inferred torque in runs A, B, and C was not accurate or that the torque was inaccurately measured in Run D. Otherwise, the necessary changes to the program should exclusively change the mass flow results.

Accounting for thermal expansion was an addition to the program, and it assumes no physical issues with the turbine test setup. Increasing turbine inlet pressure by 4% or increasing the nozzle exit diameter (DNE) and throat diameter (DNT) by 0.0007” both assume something in the turbine test setup was not as expected. Although unlikely, the current placement of the inlet pressure measurement device could be yielding measurements 4% lower than expected. The increase of 0.0007” for DNE is actually within the DNE tolerance on the mechanical drawing. The DNT tolerance is only 0.0002,” but if any corrosion from the constant presence of steam had occurred, there is a possibility that DNT would corrode enough to widen 0.0007” in diameter.
Chapter 4

Conclusions

Summary

The intended outcome of this research was to have a modern, accurate turbine modeling tool available to ARL engineers for future research applications. After translating the program from FORTRAN to MATLAB and verifying the translation, a sensitivity analysis of key subroutines was performed. The data was analyzed by comparing the MATLAB program’s results with past experimental data, then determined a number of potential problem areas in the code. Some possible fixes included accounting for thermal expansion of the nozzle, increasing the inlet pressure, and increasing the nozzle diameters at the throat and the exit. Although the modeling program is still not entirely accurate, it is now in a more functional code language, and insights into possible problems have been drawn.

List of Technologically Important Findings

1) A 4% increase in the input variable of turbine inlet pressure (TIP) yields accurate mass flow results and increases torque results from the program.

2) An increase of 0.0007” at the nozzle exit diameter (DNE) and nozzle throat diameter (DNT) parameters yields accurate mass flow results and increases torque results from the program.
3) Accounting for thermal expansion of the nozzle throat diameter (DNT) yields accurate mass flow results and increases torque results from the program.

**Future Work Suggestions**

The recommended future work for ARL engineers include: 1) to verify the dimensions of the turbine nozzles and determine if tolerances are being met; 2) to analyze the corrosion properties of the nozzle material as well to see if the nozzle diameter increases over time as a result of corrosion; 3) to investigate the accuracy of the inlet pressure measurement devices used in runs; 4) to research in greater depth both the areas of the program code that are not documented and the accuracy of steam properties used in this analysis (this research was inconclusive in determining the origin of the equations in a few lesser subroutines); and finally, and perhaps most importantly, 5) to research the assumptions used to determine the inferred torque on test runs that do not measure torque directly.
Appendix A

Team

Those who aided me with this thesis included:

   Eric White
   Caroline Rice
   Jeremy Walter
   Ken Freeman
Appendix B

Tools

The tools used to carry out this thesis research included (all on a Dell desktop computer running a Windows 7 operating system):

MATLAB
Microsoft Excel, Powerpoint, and Word
Notepad++
Mendeley
FORTRAN language and compiler
Appendix C

Broader Impacts

Economic/Manufacturability/Technology Transfer

Contributing a more accurate model for performance at low turbine exhaust pressures will save ARL a significant amount of money because fewer physical experimental tests will be required. A more accurate model means that more simulated testing can occur with trusted results.

Environmental/ Sustainability

Each time a physical test is required, resources are used. More effective modeling will reduce this need and reduce emissions from all of the elements included in performing a physical test.

Ethical

The end result of this research is the design of undersea power systems for the United States Navy. The ethical implications are that these vehicles may result in danger to both human lives and marine habitats. As such, proper design is required to ensure that these vehicles work properly and affect only what their operators intend.
Health and Safety

Accurate performance evaluation of the undersea power systems are needed to ensure the effectiveness of the final product. Since United States servicemen and servicewomen rely on these technologies, providing highly reliable products with known effectiveness are necessary for the safety of those men and women.

Social/Political/Global

ARL’s undersea power systems work ultimately results in strengthening of the United States Navy. Such improvements will allow the United States to keep our oceans safer.
REFERENCES


ACADEMIC VITA

Zechariah (Zack) Daniel Meyer
128 October Drive, State College, PA, 16801 • (814)777-0758 • zdm2011@gmail.com

OBJECTIVE
My current goal is to work in an active team atmosphere towards a tangible goal in the field of energy while developing leadership and business skills.

EDUCATION
Pennsylvania State University 2011-2015 (Expected) University Park, PA
Schreyer Honors College
Minor, Engineering Entrepreneurship
B.S., Engineering Science (Honors Degree)
B.S., Energy Engineering
• Dean’s List every semester

EMPLOYMENT
Applied Research Laboratories July 2011- Current State College, PA
Undergraduate R&D Engineer, Torpedo & Power Systems and Energy Science & Power Systems
• Performed tasks in drafting, programming, databases, test plans, and failure analysis
• Completed undergraduate thesis: research entailed analyzing turbine performance predictor methods in an extensive MATLAB program

Gentzel Corporation March 2010 – Current State College, PA
Corporate Secretary
• Aided company president in making executive decisions
• Managed and signed off on corporate documents

LEADERSHIP EXPERIENCE
President of Tau Beta Pi, PA Beta Chapter (National Engineering Honor Society)
• Doubled officer involvement, increased membership by 25%+, and began re-establishment of Tau Beta Pi involvement and recognition on campus
• Updated record-keeping spreadsheets with better formulas
• Instituted clear officer duties and more efficient policies to streamline events
• Established a new cloud-based file management system to for better organization

Eagle Scout
• Transformed a run-down playground into a spiritual prayer garden for a church
• Directed over 20 volunteers and nearly 400 man-hours of labor

IM Sports Captain
• Captain of Basketball, Volleyball, and Flag Football teams at Penn State
• Led IM Volleyball team to be league champions

SKILLS
Programming: MATLAB, Labview, Excel; some Java, C++, and HTML
CAD drafting: AutoCAD, Solidworks, and Siemens NX (Unigraphics)
Office software: Basic image editing, file management, and Microsoft Office
Foreign culture experience: 5 years of French education and practice; 5 weeks of study abroad in Italy