Justin Semchak

Linkedin https://www.linkedin.com/in/justin-semchak-200bb0188/
Email Jsemchak@gmail.com
Phone 661-437-7663

Portfolio as of February 2023

Variable size Venturi Injector Test Bench – Mazzei Injector Corp.

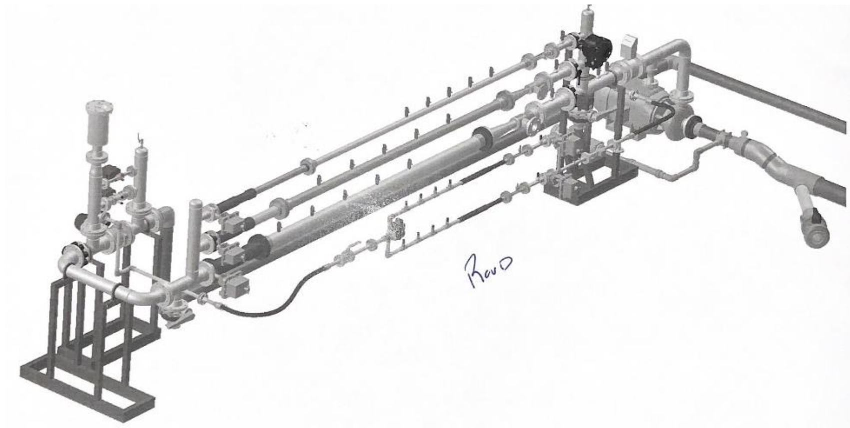

Fig. 1 ISO view, Operator side of test bench

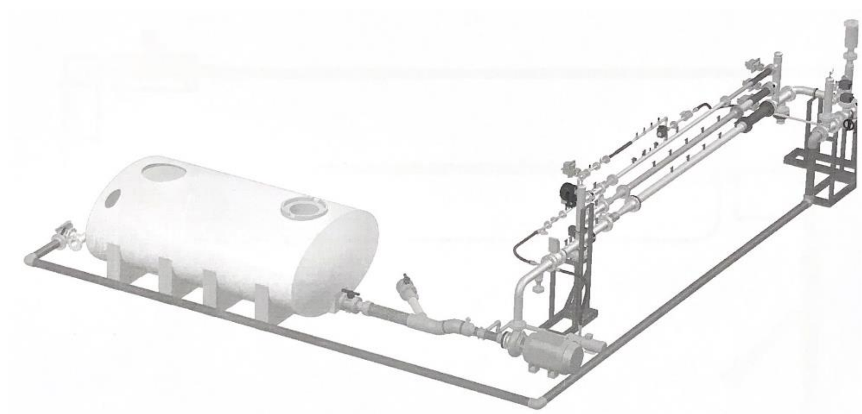

Fig. 2 ISO view, Service side of test bench

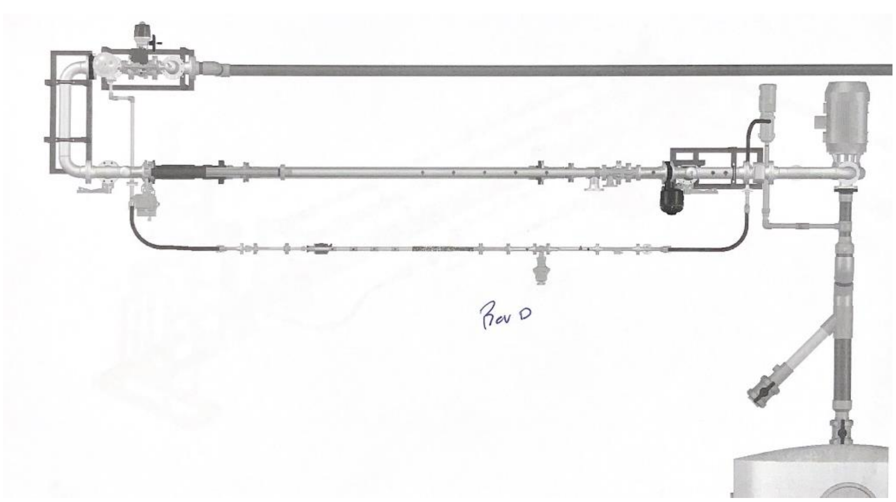

Fig. 3 Top view, Total assembly

Project Description

The main purpose of this standalone system was to have one, entirely automated, all-encompassing test bench to test Venturi injectors with a water fluid flow and air injection, and injection skids for functionality and reliability. Up until the point of this system's design, the company has struggled with two separate systems that were unreliable and inaccurate. One test bench would handle ½ inch up to 1 ½ inch injectors, while the other would handle 2, 3, and 4 inch injectors. Having two separate systems increased data collection and setup time, and extended batch testing times for the operator nearly two fold. During my internship, I had to use the current testing methods, then using SOLIDWORKS develop a system to begin to prototype an integrated, two circuit test bench to house and support all testing for the company. This system consisted of 700 unique components and was a very complex CAD project that taught me many skills in SOLIDWORKS.

Technical Details

The test bench consisted of two integrated systems that were driven by 2 separate VFD driven pumps, and a programmable logic computer or PLC. The smaller ½ inch to 1 ½ inch injector circuit would use a smaller pump for lower flow rates, and the larger circuit had a much larger pump that could support up to 4 inch injectors, as well as FAT testing for smaller skids. The test bench would approximately have a footprint of 100ft x 100ft and have a 2,000 gallon tank to support any size skid or any flow rate. The main goal of the test bench is to provide an accurate rate of injection based on downstream pressures at prescribed lengths of pipe, inlet pressures, and flow rate through the system. Being able to adjust inlet and outlet pressures with minimal internal turbulence in the pipe flow was crucial to accurate and effective testing of all systems and injectors. All valves used electric actuators that would control all inlet and outlet pressures, as well as system control via the integrated PLC.

Implementation

During my time with the company, my main goal was to design the system in SOLIDWORKS then have the fabrication division assemble the system after my 3 month internship with the team. Half of the system was operational before my departure and was a successful prototype. The test bench finally allowed the team to digitally log data at a scale not seen before, as this system was entirely automated and did not require an operator to manually record the results.

Department of energy Collegiate wind competition – Wind Turbine Development Lead

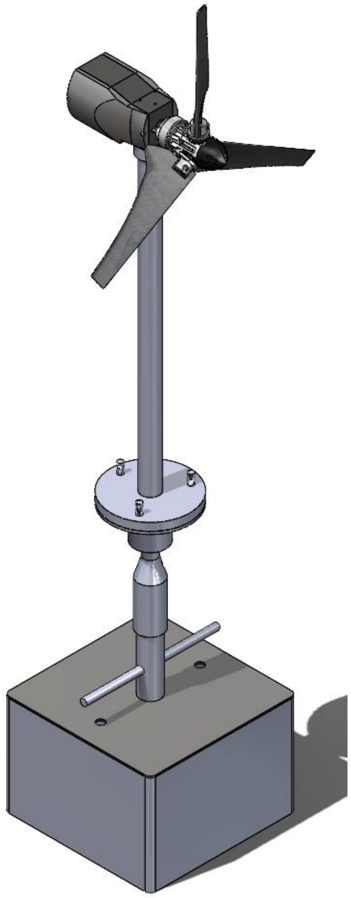

Fig. 4 ISO view, Complete Assembly

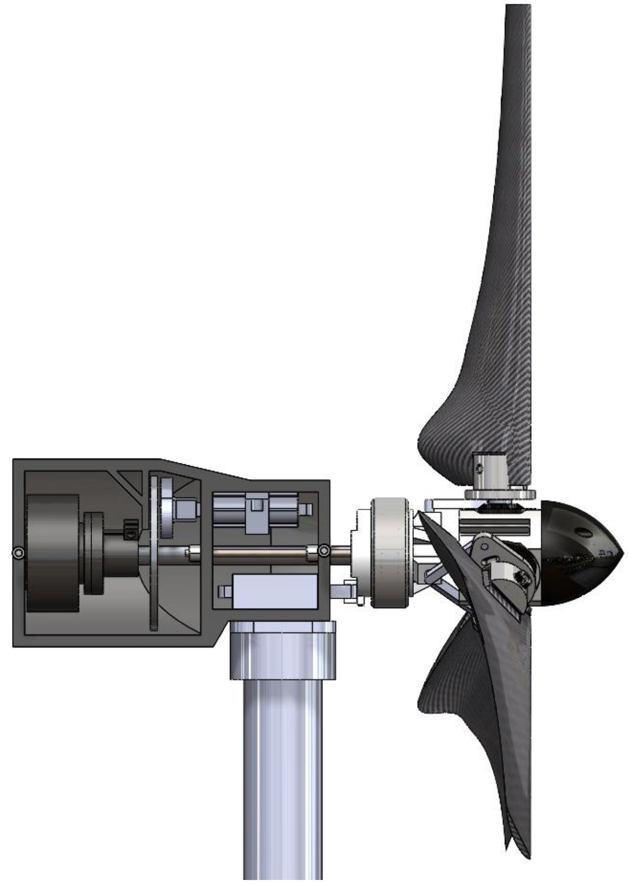

Fig. 5 Right view, Nacelle interior along with pitch, braking, and generation sub-systems

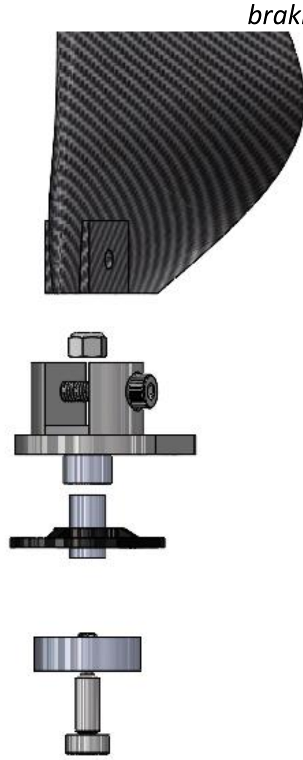

Fig. 6 Exploded view of Blade-Pitch subassembly

Project Description

For my Capstone project at Northern Arizona University, I am competing in the 2023 CWC competition. This competition challenges students to build an off shore, small-scale wind turbine that can survive a and handle a wide array of real world scenarios such as a runaway event with up to 25 m/s winds, as well as electrical failure tests. For 2023, I am the Turbine Development team lead and am managing 6 total team members to design and manufacture the wind turbine. As Turbine lead, I lead development of aerodynamics, CAD design and simulation (FEA) in SOLIDWORKS, and control state analysis using and Arduino and custom printed PCBs.

Technical Details

The max rotor diameter is 45 cm x 45 cm and the maximum area to be used by the anchorage system is 30 cm x 30 cm. To develop the aerodynamic behavior of the turbine, I used BEM theory as well as the simulation suite QBlade to simulate the chosen parameters for our blades. QBlade supported both rotor and turbine simulation, as well as blade optimization via Schmidt optimization algorithms and airfoil simulation. This allowed me to build power curves to allow design of the generation system, and the pitch systems. For the competition, there is a high speed wind test, that tests the safety and reliability of the wind turbine. With onboard control electronics, we can pitch the blades to feather, as well as engaging a disc brake system to bring the turbine to a stop in a runaway event. The onboard control electronics are based on the computation of an Arduino and utilizes a MOSFET based rectification system for the three phase power produced by the AC brushless motor. Using the RPM interpolated from the generator output voltage, we can create control states, or regions where we can adjust the pitching system to optimize the flow regime across the rotor in any given situation.

Implementation

As the competition has not approached yet, the team and I are still in the prototyping and iterating phase and have begun testing on the electrical components using a dynamometer to simulate wind induced rotation of the rotor to build a power curve of our chosen generator. The blades are being printed from a nylon and carbon fiber 3D printer filament and allows the turbine blades to achieve low weight and high bending strength and centrifugal strength.

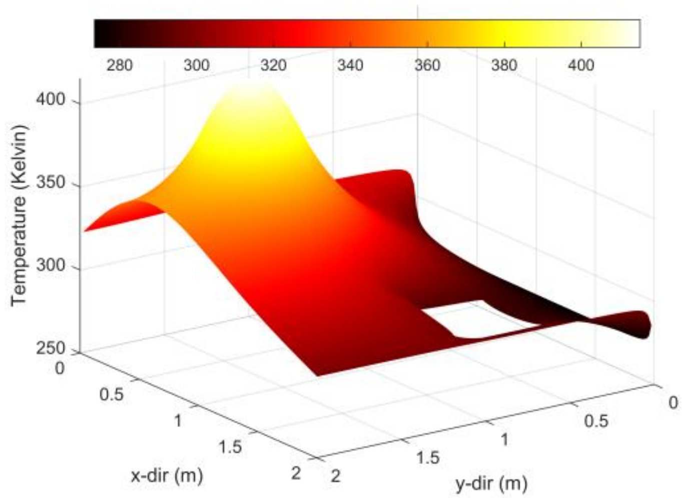

Fig. 7, 3D surface plot of temperature distribution

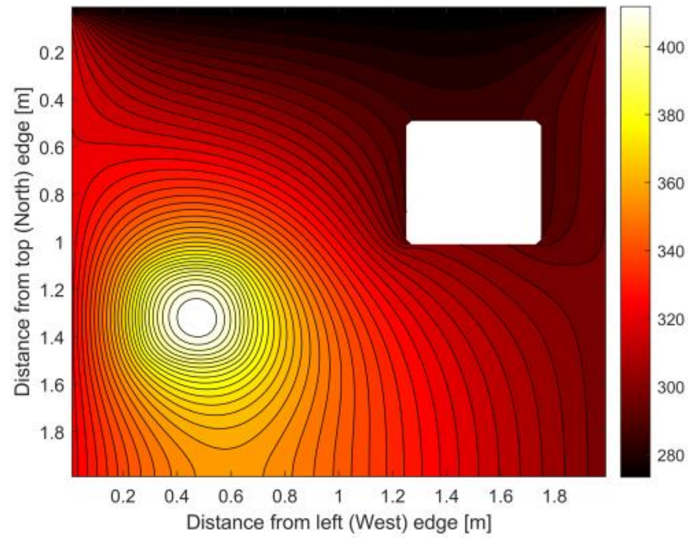

Fig. 7, 2D contour plot

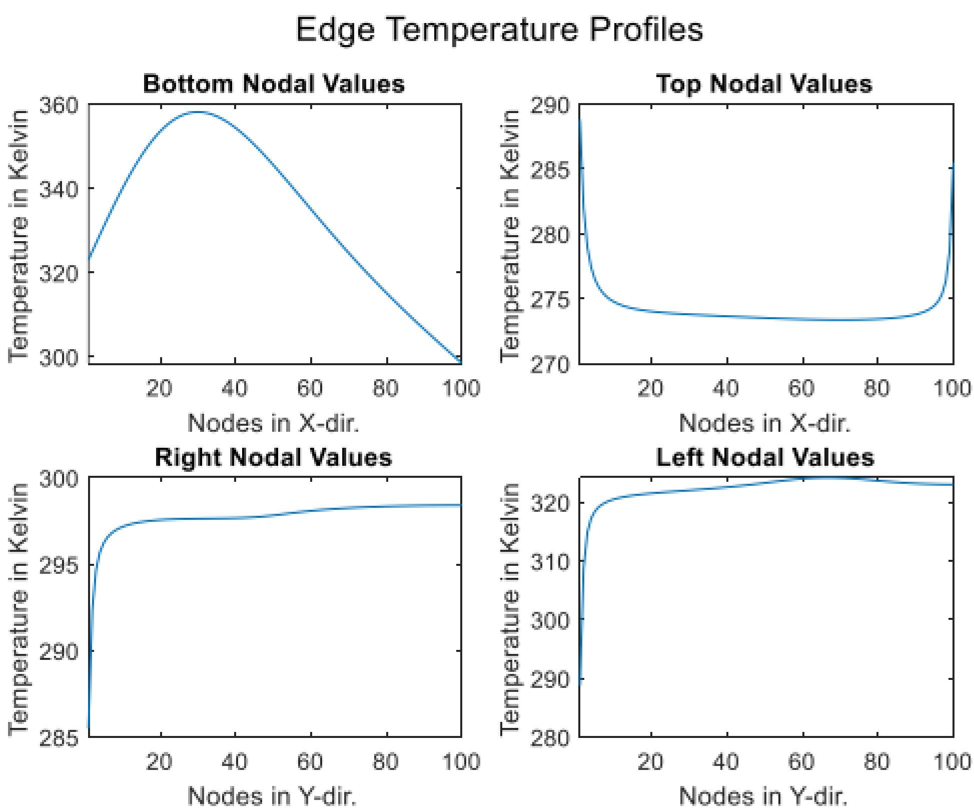

Fig. 8, Edge Temperature profile plots

Project Description

The numerical project for ME 450 was assigned to give students a chance to apply their conceptual understanding of heat transfer and related fields to a realistic and in-depth problem. The project asks students to simulate, both numerically and analytically, heat flow in a 2D square plate of material using finite difference techniques that were taught using MATLAB.

Technical Details

To begin solving this problem, I had to break down the problem geometry and other variables to apply matrix multiplication techniques for the finite difference equations. I began to analyze and discern 15 unique node types in this problem, then derive the finite difference equations for each unique node. Following this I created a meshgrid to help me locate the important locations, such as edges, in the problem geometry. Knowing the locations and dimensions of the unique nodes, I can now apply my derived finite difference equations. I created a series of if statements that if a node was of a certain type (and stored these node types in a matrix to help with debugging), carry out these calculations. The output of these equations populated a matrix A, a matrix b, and a matrix t. For the matrix multiplication used for this technique, I used MATLAB's built-in matrix inversion solver to solve for T, our temperature distribution. However, the matrix T is a column vector, so to apply this data to our square matrix (our meshgrid) I used MATLAB's reshape function to transform the column vector into a matrix of node length the problem required (Nx). After finding T, I plotted the distribution using both contour and surface plots to create a visual representation of the temperature distribution

Implementation and Code Verification

To begin verifying the validity of the code's results, we can easily visually check to see if the data is representative of the boundary conditions given in the context of the problem. For the east, west, and north edges of the problem, constant temperatures of T1 = 25 C, T2 = 49 C, and T3 = 0 C, respectively, are maintained. These boundaries are Dirichlet conditions. As seen in the problem geometry, the north boundary of 0 C (273K) will see the most heat transfer, intuitively, because of its very low temperature with respect to the rest of the node temperatures. This is seen when numerically verifying the code results. This result is also seen in the 3D surface plot as the north boundary area has the coolest temperatures of the entire block. On the south boundary of the problem, the edge is fully insulated, and therefore no heat transfer can occur. This type of boundary condition is called a Neumann condition, where dT/dx @ x = 0 = 0, because the surface is adiabatic. The south boundary can be seen as adiabatic in the 3D surface plot as the area local to the southern boundary experiences some of the most consistent highest temperatures of the entire block. To see if the numerical verification was correct, we can sum heat flows, and heat generation, to see if the heat is conserved. Theoretically, the error should be 0 when all the heat flows are summed.