

ABSTRACT

Title of Thesis: Development of a Toroidal Propeller
for Marine Applications

Daniela Colombi
Julian Cooper
Benjamin Greenberg
Nicolas Lei
Catherine Stock
Varun Unnithan

Thesis Directed by: Dr. Johan Larsson
Department of Mechanical Engineering

Toroidal propellers have demonstrated promise for increased efficiency and reduced cavitation in an aerial environment, particularly in small-scale applications such as unmanned aerial vehicles. However, there is limited research on their performance in a marine environment. To address this gap, we developed a methodology to design marine toroidal propellers with an expanded area ratio (EAR) equivalent to that of a traditional screw propeller. The toroidal propeller geometry is parametrically varied by blade reference line and blade angle modifications to attempt to achieve an efficiency identical to the screw propeller. Using computational fluid dynamics (CFD), the baseline screw propeller and toroidal propeller geometries were simulated in identical marine environments, which were validated against experimental data. Thrust, torque, and efficiency were compared to assess effective parametric variation. This provides a process for evaluating toroidal propeller applications and contributes to a better understanding of how toroidal propellers behave in marine environments. CFD results revealed that toroidal propellers require a higher torque and produce a higher thrust than traditional screw propellers. A careful variation of propeller parameters is needed to balance torque and thrust for a noticeable increase in efficiency.

Development of a Toroidal Propeller for Marine Applications

by

Daniela Colombi, Julian Cooper, Benjamin Greenberg,
Nicolas Lei, Catherine Stock, Varun Unnithan

Thesis submitted in partial fulfillment
of the requirements of the Gemstone Honors Program.
University of Maryland
2026

Advisory Committee:

Dr. Johan Larsson, Advisor
Dr. Cecilia Huertas Cerdeira, Discussant
Dr. Kenneth Kiger, Discussant
Dr. Ethan Lust, Discussant
Dr. Peter Sunderland, Discussant
Mr. Matthew Cain, Librarian

© Copyright by

Daniela Colombi, Julian Cooper, Benjamin Greenberg,
Nicolas Lei, Catherine Stock, Varun Unnithan

2026

Acknowledgments

We would like to thank Ishaan Bhardvaj, Neel Jay, Malcolm Maas, Teja Nalagorla, Jordan Qureshi, Yin Wang, and Daniel Yuan for their assistance and the time they dedicated to supporting our team. We would also like to thank our librarian, Mr. Matthew Cain, for his guidance and support during our research process. We would like to thank the gemstone staff for teaching us the research process and always being a source of guidance. We would like to thank our discussants for providing feedback. Finally, we would like to express our appreciation to our mentor, Dr. Johan Larsson, for his mentorship, input, and mostly successful, yet continuous, effort of keeping us on track the past four years.

Contents

1	Introduction	1
2	Propeller Geometries	5
2.1	NACA Series Airfoils	5
2.2	Screw Propellers	7
2.3	Toroidal Propellers	9
2.3.1	Frenet-Serret Method	10
2.3.2	Ye et al. Method	18
2.4	4119 and BAR Matching	23
2.4.1	Screw Propeller 4119 Parameters	23
2.4.2	Blade Area Ratio Matching	24
3	Analysis Methods	28
3.1	CFD	28
3.2	Propeller Simulations	31
3.2.1	Screw Propeller	31
3.2.2	Toroidal Propeller	38
3.2.3	Final CFD Simulation	40
4	Optimization of Toroidal Propellers	43
4.1	Initial Propellers	43
4.2	Manual Optimizations	45
4.2.1	Wheelbase Distributions	46
4.2.2	Skew Distributions	50
4.2.3	Pitch Distributions	54
5	Discussion	66
A	Github	71
B	Educational Video	72
	References	73

List of Figures

1	Marine Screw Propeller Patent [7]	2
2	1888 Toroidal Propeller Patent [6]	3
3	Sharrow Engineering LLC Toroidal Propeller Patent [10]	4
4	Geometry of NACA airfoils. Reproduced from Abbott, Von Doenhoff, and Stivers (1945) [12].	5
5	Screw Propeller Geometry [14]	8
6	Example centerline with the ribs being airfoils placed perpendicularly, with no local transforms, along the centerline, showing how the Frenet-Serret method works to generate a blade. Only 10 airfoils along the blade are shown for visualization.	11
7	Visualization of the parameter t as it goes along the parameterized airfoil.	12
8	Effect of different local airfoil transforms on the cross-section created.	13
9	An example blade generated from given local airfoil transform functions parameterized by s	14
10	Representation of the coordinate system that is used to define centerline control points.	15
11	Point cloud with airfoil profiles and guide curves of propeller defined in [15].	19
12	Intersection locations during loft operation of propeller defined in [15].	20
13	Point cloud with airfoil profiles and guide curves of modified [15] propeller as per Table 2.	23
14	Successful geometry creation of modified [15] propeller as per Table 2.	23
15	Successful geometry creation of the 4119 propeller as per Table 5 and Table 6.	24
16	Point cloud with airfoil profiles and guide curves of TORUS 0000 propeller as per Table 8 and Table 9.	27
17	Successful geometry creation of TORUS 0000 propeller as per Table 8 and Table 9.	27
18	Meshes created in STAR-CCM+ to illustrate types of meshes.	29
19	Pressure coefficient graph of NACA0012 airfoil.	30
20	Velocity magnitude vector graph.	30
21	Pressure distribution along a NACA0012 with 5° angle of attack.	31
22	Domain regions with inlet and outlet locations.	32
23	Dimensions of fluid regions.	32
24	Mesh of full domain and rotating region.	33
25	Y^+ values along the front surface and back surface of a 4119.	33
26	CFD simulation results compared to open-water performance curves [30].	34
27	Double density mesh of full domain (a) and rotating region (b).	34
28	Half density mesh of full domain (a) and rotating region (b).	35

29	Comparison of results from mesh and iteration dependence study for screw propellers.	36
30	Half density mesh v2 of full domain (a) and rotating region (b). . . .	37
31	Relative error between base mesh and the three other versions.	38
32	Comparison of results from mesh and iteration dependence study for toroidal propellers.	39
33	Relative error between base mesh and the three other versions.	40
34	yz-plane section cut of toroidal propeller simulation's rotating region showing resolutions of half density version two (a) and new refined region (b).	40
35	CFD results of the 4119 propeller run at design conditions.	41
36	Velocity in the z-direction shown on the yz-plane.	41
37	Pressures at two radii of the 4119.	42
38	Front views of the 4119 and the TORUS 0000 toroidal propeller. . . .	43
39	Pressure across the blades of the 4119 and the TORUS 0000 toroidal propeller.	44
40	Pressures at 0.7 and 0.9R for the TORUS 0000.	45
41	TORUS 1000 propeller as per Table 13.	48
42	Pressures at 0.7R and 0.9R for the TORUS 1000.	48
43	TORUS 2000 propeller as per Table 13.	49
44	Pressures at 0.7R and 0.9R for the TORUS 2000.	49
45	TORUS 0100 propeller as per Table 14.	51
46	Pressures at 0.7R and 0.9R for the TORUS 0100	51
47	TORUS 0200 propeller as per Table 14.	52
48	Pressures at 0.7R and 0.9R for the TORUS 0200.	52
49	TORUS 1100 propeller as per Table 14.	53
50	Pressures at 0.7R and 0.9R for the TORUS 1100.	53
51	TORUS 2010 propeller as per Table 15.	57
52	Pressures at 0.7R and 0.9R for the TORUS 2010.	57
53	TORUS 0020 propeller as per Table 15.	58
54	Pressures at 0.7R and 0.9R for the TORUS 0020.	59
55	TORUS 1020 propeller as per Table 15.	60
56	Pressures at 0.7R and 0.9R for the TORUS 1020.	60
57	TORUS 1120 propeller as per Table 15.	61
58	Pressures at 0.7R and 0.9R for the TORUS 1120.	62
59	TORUS 2020 propeller as per Table 15.	63
60	Pressures at 0.7R and 0.9R for the TORUS 2020.	63
61	TORUS 1031 propeller as per Table 15 and Table 16.	64
62	Pressures at 0.7R and 0.9R for the TORUS 1031.	65
63	Comparison of all TORUS propellers and the 4119	67
64	Pressure on both sides of TORUS 0100.	68
65	Pressure on both sides of TORUS 2010.	69
66	Pressures at 0.7R and 0.79R for the tipless TORUS 1100.	70

List of Tables

1	NACA66 a=0.8 TMB Mod Data (Y_t/t , Y_c/f , and $(dY_c/dx)/f$). Adapted from [13]	7
2	Select Blade Parameters of [15] and Modifications	21
3	Blade Parameters of [15] with Modifications	22
4	Blade Parameters of [15]	22
5	4119 Blade Parameters. Adapted from [24].	24
6	4119 Propeller Parameters. Partially adapted from [24].	24
7	c/D smoothing approximation variables.	25
8	TORUS 0000 Blade Parameters	26
9	TORUS 0000 Propeller Parameters	26
10	Number of cells in each mesh density.	35
11	Run conditions for the 4119.	41
12	CFD Performance Comparison of 4119 and TORUS 0000	43
13	Various Wheelbase Distributions	47
14	Various Reference Line Skew Distributions	50
15	Various Pitch Distributions and Geometric Angle of Attack Multiplier	55
16	Various Vertical Angle Distributions	56
17	Performance of Designed Propellers	66

Nomenclature

α	Local Vertical Angle
β	Local Angle of Attack
η_0	Propeller Open-water Efficiency $\eta_0 = \frac{JK_T}{2\pi K_Q}$
ν	Kinematic Viscosity of Water
ψ	Local Roll
ρ	Density of Water
θ_s	Local Additional Skew
φ	Local Reference Skew
c	Local Chord Length
c_{LE}	Local Chord Offset from the Leading Edge
D	Propeller Diameter
f	Local Camber
J	Advance Ratio $J = \frac{V_{inlet}}{nD}$
k	Local Additional Rake
K_Q	Coefficient of Torque $K_Q = \frac{Q}{\rho n^2 D^5}$
K_T	Coefficient of Thrust $K_T = \frac{T}{\rho n^2 D^4}$
L	Propeller Wheelbase
l	Local Wheelbase
n	Rotational Speed
P	Local Pitch
Q	Propeller Torque
R	Propeller Radius
r	Local Radius
Re_r	Reynolds Number at Local Radius $Re_r = \frac{c_{0.7} \sqrt{V_{inlet}^2 + (0.7\pi nD)^2}}{\nu}$
s	Airfoil Station

T	Propeller Thrust
t	Local Thickness
V_{inlet}	Inlet Velocity
Z	Number of Blades

1 Introduction

The United Nations Conference on Trade and Development reports that maritime shipping carries 80% of internationally traded goods, more than any other mode of transportation [1]. The abundance of maritime shipping makes it the dominant mode of transportation not only in the United States but worldwide. Given this central role in global trade, maritime shipping plays an important role in economic development, job creation, and industrialization.

Despite the economic importance of maritime shipping, its environmental impacts are severe. In 2022, the global maritime shipping sector was responsible for roughly 2% of total greenhouse gas emissions. Despite ongoing legislative and research initiatives, this share is expected to continue rising in the coming years [2]. Improving the propulsion efficiency of freighters can reduce fuel consumption and emissions.

The primary element of a ship’s propulsion system is the propeller, whose efficiency is dependent on its geometry. The blades of a propeller act as hydrofoils, producing a difference in pressure on either side. This pressure differential generates a force that propels the ship. However, this can be problematic if the local pressure near the blade becomes low enough, small vapor bubbles form and implode violently in a process known as cavitation [3]. This results in damage to the propeller and loss of efficiency. In order to prevent this, the pressure differential—and therefore the propulsive force—must be distributed over a large area to decrease low pressure areas. [4].

The risk for cavitation can be reduced by adjusting a propeller’s geometric parameters. A propeller is characterized by a series of cross-sectional airfoils arranged at specific radial positions centered around the hub. This method of propeller generation allows the propeller to be defined by physical characteristics such as pitch, skew, and rake. By modifying these characteristics, the amount of thrust produced, torque needed, and overall efficiency can be changed and optimized.

Ships are predominantly equipped with screw propellers as a means of propulsion, a typical example of which is shown in Figure 1. An alternative is the toroidal propeller, which features blades that loop back into the hub, forming a continuous toroidal shape. The first toroidal propeller dates back to the late 1800s, designed by Charles Myers, working with Fawcett, Preston and Company [5] Myers aimed to increase thrust by reducing the vacuum at the blade tips [6].

April 4, 1961

O. A. WIRKKALA
MARINE PROPELLER
Filed Feb. 4, 1958

2,978,040

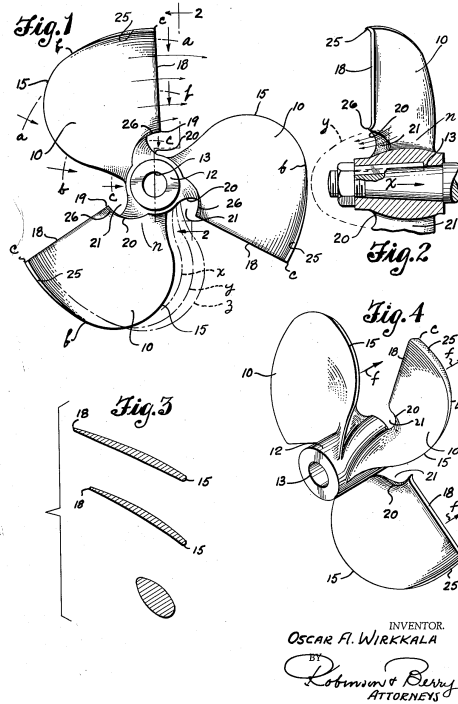


Figure 1: Marine Screw Propeller Patent [7]

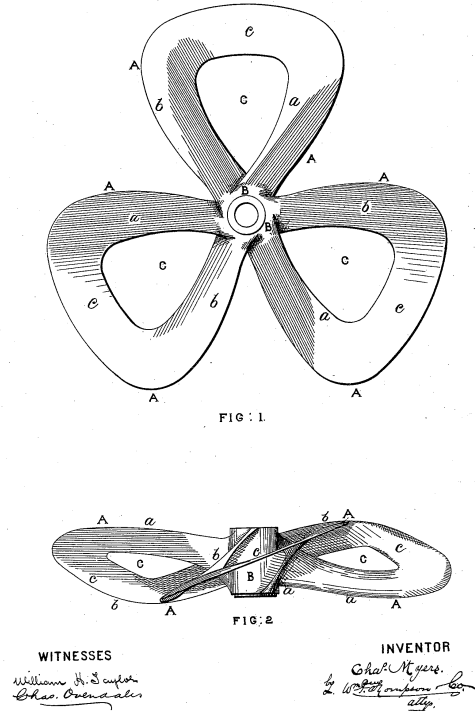


Figure 2: 1888 Toroidal Propeller Patent [6]

An experimental comparison was conducted between two large twin-screw ferry boats, the *Mersey* and the *Wiral*, which were identical except that one vessel was fitted with a Myers-type propeller. The results indicated a noticeable reduction in vibration when using the toroidal propeller [8]. Although engineering practices in the late nineteenth century lacked the advanced measurement and analysis tools available today, historical observations still provide useful insight. For example, reports for the *Arrow* claim that its towing capacity increased from approximately eight barges to eleven, suggesting a meaningful performance improvement [5].

Over the years, toroidal propeller development has seen limited adoption, periodically reemerging in different forms. One notable resurgence occurred in the late 1960s, when David B. Sugden patented annular, closed-loop rotor concepts for marine propulsion [9]. While these designs differ from modern implementations, they reflect continued interest in reducing losses associated with conventional propeller tips. More recently, in 2018, Sharrow Engineering LLC filed a patent (Figure 3) for a toroidal propeller design aimed at high-speed boats. According to Sharrow, this geometry can improve fuel efficiency and overall performance in high-performance marine vessels [10].

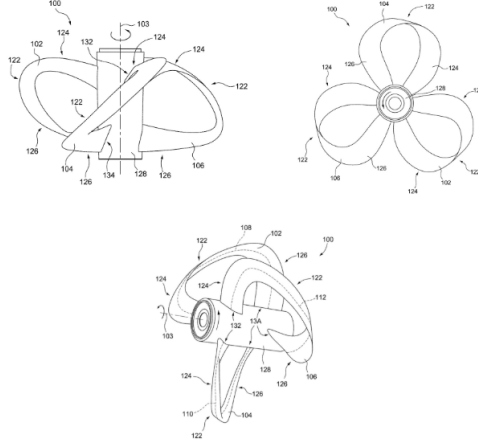


Figure 3: Sharrow Engineering LLC Toroidal Propeller Patent [10]

In 2022, MIT Lincoln Laboratory patented a similar toroidal propeller concept for use in unmanned aerial systems. Their accompanying research demonstrated reductions in the radiated acoustic noise, suggesting that the geometry mitigates tip-vortex formation and thus possibly cavitation when used in liquids [11]. These mechanisms may also explain the increased fuel efficiency observed in marine applications of the Sharrow propeller.

This thesis aims to develop a methodology for comparing conventional screw propellers with unconventional toroidal designs. Using computational fluid dynamics (CFD) and parametric optimization, the proposed framework evaluates a traditional screw propeller and a toroidal propeller with matched expanded area ratio (EAR). Using the screw propeller as a baseline, the toroidal propeller blade reference line and blade angle is parametrically varied to determine if its efficiency can match or exceed that of its parent screw propeller.

2 Propeller Geometries

This work considers both traditional screw propellers and toroidal propellers. The geometries of both rely heavily on the NACA series of airfoil cross sections. The toroidal propellers developed in this project were compared to a specific screw propeller, the Carderock 4119, with toroidal propellers having the same expanded aspect ratio.

2.1 NACA Series Airfoils

4-Series

The NACA 4-series airfoil is defined in [12]. The mean camber line $y_c(t)$ is defined piecewise:

$$y_c(t) = \begin{cases} \frac{m}{p^2}(2pt - t^2), & 0 \leq t \leq p, \\ \frac{m}{(1-p)^2}[(1-2p) + 2pt - t^2], & p < t \leq 1. \end{cases} \quad (1)$$

The thickness distribution $y_t(t)$ is given by:

$$y_t(t) = 5 \text{ thickness} \left[0.2969\sqrt{t} - 0.1260t - 0.3516t^2 + 0.2843t^3 - 0.1036t^4 \right]. \quad (2)$$

where:

- t is the normalized chord length parameter ($0 \leq t \leq 1$),
- m is the maximum camber,
- p is the location of maximum camber,
- thickness is the maximum thickness as a fraction of the chord.

As all NACA airfoils, the upper and lower surfaces are generated by offsetting the thickness perpendicular to the camber line. These equations are also shown in Figure 4.

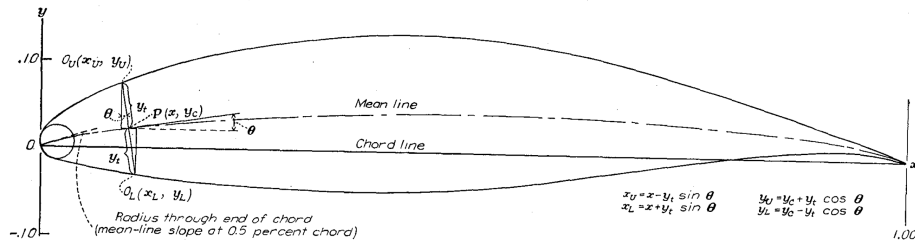


Figure 4: Geometry of NACA airfoils. Reproduced from Abbott, Von Doenhoff, and Stivers (1945) [12].

$$x_u(t) = x - y_t(t) \sin(\theta_c(t)), \quad (3)$$

$$y_u(t) = y_c(t) + y_t(t) \cos(\theta_c(t)), \quad (4)$$

$$x_l(t) = x + y_t(t) \sin(\theta_c(t)), \quad (5)$$

$$y_l(t) = y_c(t) - y_t(t) \cos(\theta_c(t)) \quad (6)$$

where $\theta_c(t)$ is the angle of the camber line slope:

$$\theta_c(t) = \arctan\left(\frac{dy_c}{dt}\right). \quad (7)$$

6-Series

The NACA 6-series airfoil is also defined in [12], however mean camber lines are generated numerically for the percent of the chord where laminar flow is maintained. The NACA66 a=0.8 TMB Mod presented in [13] and Table 1 is highly favored in marine applications due to a larger favorable operating range. This airfoil alters the leading edge to improve manufacturability, and both the leading and trailing edge to increase the pressure range before cavitation.

Table 1: NACA66 a=0.8 TMB Mod Data (Y_t/t , Y_c/f , and $(dY_c/dx)/f$). Adapted from [13]

Station	Y_t/t	Y_c/f	$(dY_c/dx)/f$
1.0000	0.0333	0.0000	-3.003
0.9750	0.0748	0.0823	-3.441
0.9500	0.1143	0.1713	-3.668
0.9000	0.1877	0.3586	-3.768
0.8500	0.2532	0.5425	-3.523
0.8000	0.3110	0.7027	-2.712
0.7500	0.3612	0.8121	-1.813
0.7000	0.4035	0.8892	-1.295
0.6500	0.4383	0.9434	-0.885
0.6000	0.4653	0.9786	-0.532
0.5500	0.4846	0.9971	-0.211
0.5000	0.4962	1.0000	0.091
0.4500	0.5000	0.9881	0.385
0.4000	0.4952	0.9615	0.678
0.3500	0.4832	0.9202	0.933
0.3000	0.4637	0.8635	1.292
0.2500	0.4363	0.7905	1.635
0.2000	0.4000	0.6993	2.023
0.1500	0.3521	0.5869	2.488
0.1000	0.2907	0.4482	3.100
0.0750	0.2525	0.3657	3.515
0.0500	0.2066	0.2712	4.083
0.0250	0.1466	0.1586	5.023
0.0125	0.1044	0.0907	5.944
0.0075	0.0812	0.0595	6.617
0.0050	0.0665	0.0423	7.149
0.0000	0.0000	0.0000	0.000

2.2 Screw Propellers

Screw Propeller Equations

Propeller geometry is commonly characterized by the following parameters: blade number, propeller diameter, hub diameter, pitch, skew, and rake [14]. A visualization of these parameters can be seen in Figure 5.

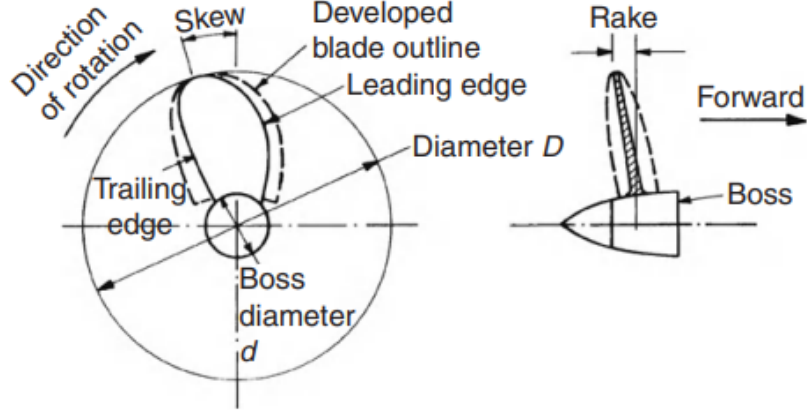


Figure 5: Screw Propeller Geometry [14]

Screw propellers have the blade parameters skew, rake, and pitch, defined on a radial basis. These three parameters, along with radius, form 3D positioning and one of the three orienting rotations for the placement of each airfoil. Pitch is the theoretical distance the propeller advances in one full rotation. It is analogous to angle of attack. It is defined in Equation 8 although can also be found with the nondimensionalized Equation 9. Traditionally angle of attack is defined as α , although for consistency with [15], β is used.

$$\tan(\beta) = \frac{P}{2\pi r} \quad (8)$$

$$\tan(\beta) = \frac{(P/D)}{\pi(r/R)} \quad (9)$$

Skew is the angular displacement around the hub from the blade reference line. Rake is the linear displacement along the hub from the blade reference line [14][16]. Rake can also be induced by applying a skew before the pitch rotation (θ_s). The screw propeller reference line defined in Equation 10 is a line perpendicular to the hub. The screw generator line defined in Equation 11 is the sum of the propeller reference line and all sources of rake and skew. The blade surface is expressed in Equation 12 where s is the station of the airfoil and y is the calculated upper and lower surfaces. It is understood that while there are varying definitions about skew and rake, this work follows the definition in [15] to keep consistent with the toroidal propeller definitions.

$$\begin{cases} x = 0 \\ r = r \\ \theta = 0 \end{cases} \quad (10)$$

$$\begin{cases} x = k + r\theta_s \tan \beta \\ r = r \\ \theta = \theta_s \end{cases} \quad (11)$$

$$\begin{cases} x = k + r\theta_s \tan \beta + (s - \frac{c}{2}) \sin \beta - y \cos \beta \\ r = r \\ \theta = \theta_s + \frac{1}{r}[(s - \frac{c}{2}) \cos \beta + y \sin \beta] \end{cases} \quad (12)$$

Different sections of a screw propeller blade travel at different linear velocities, which changes the optimal angle of attack along the radius. To maintain efficient thrust generation, propeller blades are therefore twisted, with the pitch angle decreasing from root to tip to compensate for these variations in speed. In general, the blades of a propeller lie approximately on a helicoidal surface around the hub, which provides a convenient reference for defining the pitch and twist distribution along the blade [17].

A propeller can also be categorized by the blade area ratio (BAR). While there are different types of BAR [16], this work will focus on the expanded blade area ratio (EAR), which is defined as the flattened, unpitched, area of the blade divided by the total disk area. Past studies use EAR as a baseline of creating new propellers [18] and is commonly analyzed for cavitation reasons [19].

Point Cloud

Using the defining equations of the propellers, we are able to generate a list of points that can then be connected into a propeller. First, a table of blade parameters is generated. Second, a base airfoil is chosen. Third, the defining equations are solved with the relevant parameters in MATLAB. This program then writes the constituent points of each airfoil, in order, along with guide curve information to explicitly define the contours of each propeller. These guide rails connect each corresponding airfoil point across all of the airfoils. Propeller information, such as number of blades and hub radius, are also defined in this output file. Fourth, the output file is read through a custom Fusion 360 add-in and a propeller is generated. Fusion 360 reads each airfoil and guide curve as a cubic fit spline. It takes each airfoil and guide curve as an input to a surface loft function, which attempts to solve the geometry. In the special case where a screw propeller has a zero chord length tip, a tangency feature is called to refit the guide curves. If successful, the ends of the surface are closed, the hub is formed, the singular blade is patterned around the hub, and the geometry is combined. This file can then be exported as a .STEP file. If unsuccessful, the loft function throws an error (likely a self-intersection or a zero-thickness geometry) and the geometry cannot be built. Self-intersections occur when lofts attempt to bend at too sharp of a bend radius or when the guide curves pass through each other, causing the surfaces to self-intersect. If the latter case were allowed to solve, this would create a zero-thickness region at the intersection.

2.3 Toroidal Propellers

Toroidal propellers, as the name suggests, are propellers whose blades are constructed into a torus shape. Rather than coming to a sharp edge, like screw propellers, the toroidal propeller loops back to reconnect with the central hub, leaving

a hole in the middle of the blade. The idea of a propeller with a hole in the center of each of its blades dates back to 1888, when Charles Myers filed a patent for the improvement of the screw propeller [6].

Similar to our current goal, Myers aimed to increase thrust by reducing the vacuum at the blade tips [6]. Since then, numerous patents have been granted for related designs, but until recently there has been little empirical data demonstrating improvements in efficiency. Modern toroidal propeller designs with documented performance benefits include Gregory Sharrow’s design, first patented in 2013 [10], and the toroidal drone propellers patented by Thomas Sebastian and Christopher Strem in 2017 [20].

Sebastian and Strem of MIT’s Lincoln Laboratory introduced toroidal propellers for drones as a solution to the noise pollution caused by conventional rotors. Their looped geometry significantly reduced acoustic emissions by minimizing tip vortices relative to traditional blades [11]. These tip vortices are analogous to those produced by cavitation in water. Therefore, if a toroidal propeller can suppress vortex formation in air, it is reasonable to expect a similar effect underwater, potentially reducing drag and improving fuel efficiency.

Experimental data also show that toroidal drone propellers can generate greater thrust than conventional propellers of comparable size across 1,000–20,000 rpm for small multirotor platforms [21]. Because toroidal propellers for drones are a very recent development, current research on their performance and optimization remains limited. Nonetheless, other drone optimization studies acknowledge that toroidal geometries could be adapted to a range of operating conditions with further shape refinement [22]. CFD has already been used in the design of low-noise drone propellers, but the introduction of toroidal propellers suggests that substantial research is still needed to fully understand their aerodynamic capabilities [23].

Two avenues of toroidal propeller generation were explored and are outlined in the following sections. The Frenet-Serret method defines a baseline airfoil and extrudes it along a closed three-dimensional curve. The Ye et al. method modifies the conventional screw propeller equations by introducing additional orientation rotations and reference line parameters to generate a toroidal propeller. Both methods allow for parametric control of the propeller geometry. As the goal is a direct comparison to a traditional screw propeller, the Ye et al. method proved more applicable, as it allowed for direct modification of standard propeller parameters.

2.3.1 Frenet-Serret Method

An initial approach to build a toroidal propeller can be a geometric one, or a naive approach. Building the propeller geometrically offers an easily parameterizable set of symbolic equations which are constructed through 3D extrusions and transforms done through coordinate transforms. This approach builds a toroidal propeller by extruding a NACA-profile airfoil along a closed 3D curve, which will serve as the centerline for that toroidal blade. The steps are:

1. Define the 2D airfoil cross section.
2. Create a smooth centerline via NURBS.

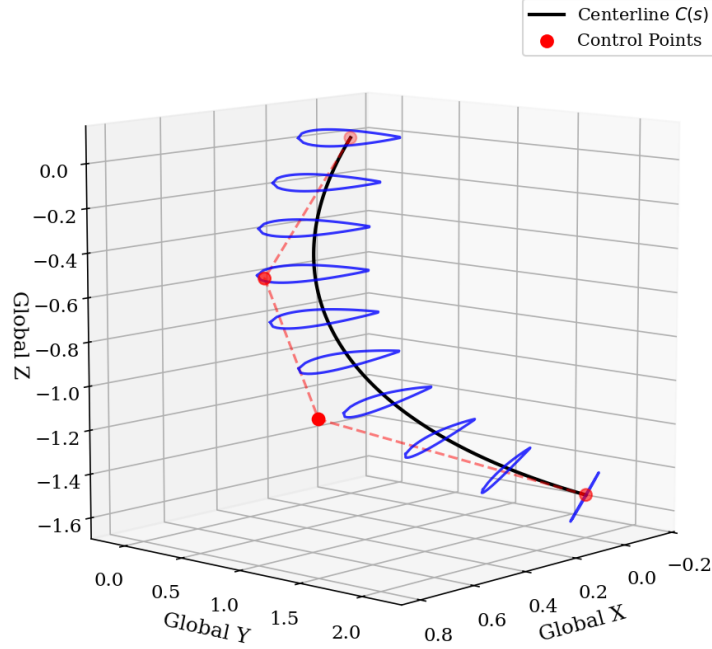


Figure 6: Example centerline with the ribs being airfoils placed perpendicularly, with no local transforms, along the centerline, showing how the Frenet-Serret method works to generate a blade. Only 10 airfoils along the blade are shown for visualization.

3. At each point on that curve, rotate and scale the cross section.
4. Assemble multiple blades around a cylindrical hub and perform a Boolean union.

Airfoil Profile

First, the 2D airfoil which will be extruded along the spline which represents the centerline of the blade needs to be constructed. For the airfoil cross-sections, the standard NACA 4-digit formulas are used. However, for the sake of simplicity and computational efficiency (particularly for the normalization of mesh sizes), $\theta_c(t)$ can be set to zero, assuming negligible camber line slope or that the model already represents a sufficient portion of the airfoil's geometry space without needing this consideration. When $\theta_c(t) = 0$, the thickness is just applied in the y -direction.

The parameter t is divided into two intervals, as seen in Figure 7.

- $[0, 1)$ for the upper surface,
- $(1, 2]$ for the lower surface (shifted to maintain continuity).

The complete airfoil coordinates are defined using Heaviside functions:

$$x_{\text{airfoil}}(t) = x_u(t) \cdot H(1 - t) + x_l(2 - t) \cdot H(t - 1), \quad (13)$$

$$y_{\text{airfoil}}(t) = y_u(t) \cdot H(1 - t) + y_l(2 - t) \cdot H(t - 1), \quad (14)$$

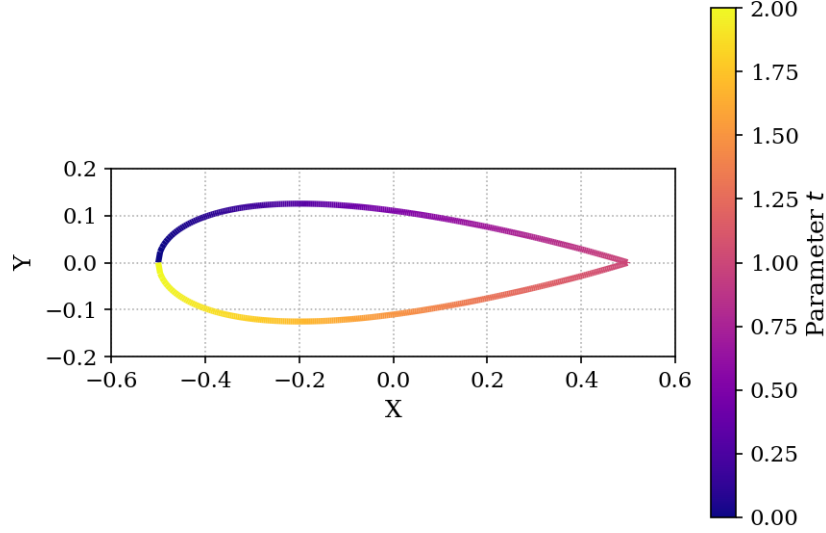


Figure 7: Visualization of the parameter t as it goes along the parameterized airfoil.

where H is the Heaviside step function. This ensures that the airfoil function, $\mathbf{f}(t) = (x(t), y(t))$, is an injective function with respect to parameter t .

To account for changes in blade orientation and cross-sectional size along the centerline, rotation and scaling transformations are applied as functions of the parameter s , which parametrizes the centerline curve. This way, as the airfoil is applied along the centerline of the blade, it can be made to change its angle of attack or scale in its local x or y coordinate directions. Since by the nature of this approach is to use the Frenet-Serret frame, which ensures that each airfoil cross section is placed to be perpendicular to the centerline, seen in Figure 6, these transformations allow for the mocking of effects like pitching or skewing the airfoil as it goes along the blade from hub to tip and back to the hub. These local effects can be seen in Figure 8.

The rotation matrix $R(s)$ for an angle of attack $\alpha(s)$ is:

$$R(s) = \begin{bmatrix} \cos \alpha(s) & -\sin \alpha(s) \\ \sin \alpha(s) & \cos \alpha(s) \end{bmatrix} \quad (15)$$

The angle of attack $\alpha(s)$ can be defined as a polynomial function of s :

$$\alpha(s) = a_\alpha s^4 + b_\alpha s^3 + c_\alpha s^2 + d_\alpha s + e_\alpha \quad (16)$$

Scaling factors $S_x(s)$ and $S_y(s)$ modify the airfoil dimensions:

$$S_x(s) = a_{sx} s^4 + b_{sx} s^3 + c_{sx} s^2 + d_{sx} s + e_{sx}, \quad (17)$$

$$S_y(s) = a_{sy} s^4 + b_{sy} s^3 + c_{sy} s^2 + d_{sy} s + e_{sy} \quad (18)$$

These functions allow for parametric scaling along the blade's path, as they are parameterized by s , the parameter for the centerline of the toroidal blade. The two

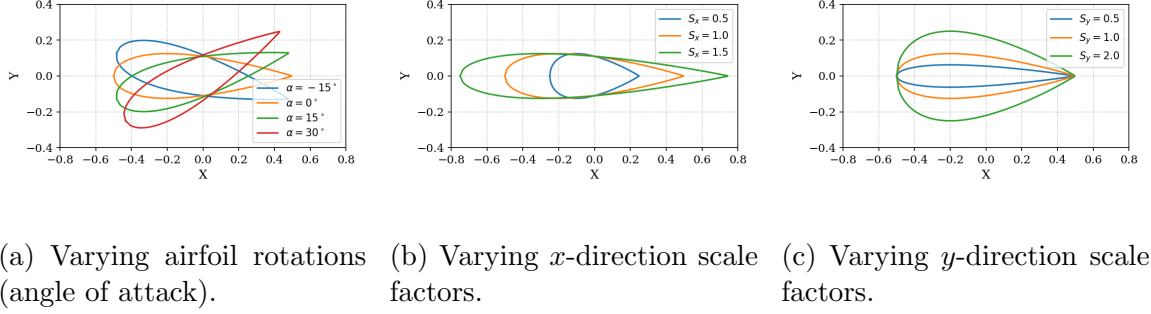


Figure 8: Effect of different local airfoil transforms on the cross-section created.

scaling functions scale the airfoil's x and y dimensions. Traditional propellers and other definitions of toroidal propellers may represent the airfoil transformations along the blade by performing rotations in all three axis. Instead, for the sake of simplicity, as well as being able to place every 2D airfoil perfectly perpendicular to the centerline, the rotation about z (the yaw) is represented with the angle of attack function, and for roll and pitch rotations, the scaling functions are used as a proxy. This is because for any roll or pitch, you can represent it by its projection on the plane perpendicular to the centerline by simply scaling either the x or y dimension, respectively, by $\cos(\phi)$, where ϕ is the angle to have been rotated by in that axis.

The transformed airfoil coordinates are:

$$\begin{bmatrix} X_{\text{rotated}}(s, t) \\ Y_{\text{rotated}}(s, t) \end{bmatrix} = R(s) \cdot \begin{bmatrix} x_{\text{airfoil}}(t) \\ y_{\text{airfoil}}(t) \end{bmatrix} \quad (19)$$

Scaling is applied as:

$$X_{\text{scaled}}(s, t) = S_x(s) \cdot X_{\text{rotated}}(s, t), \quad (20)$$

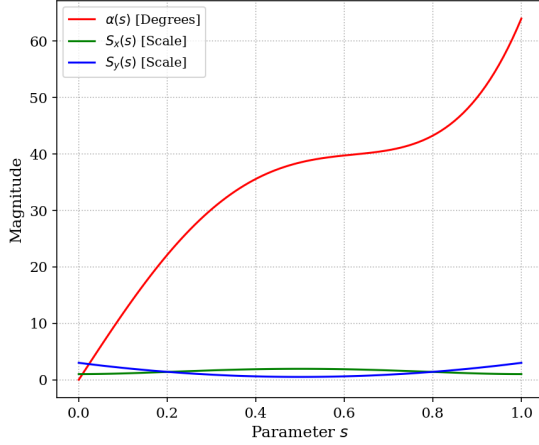
$$Y_{\text{scaled}}(s, t) = S_y(s) \cdot Y_{\text{rotated}}(s, t) \quad (21)$$

In designing the transformations for angle of attack, x -scale, and y -scale along the blade centerline, it is essential to select function forms that can model a wide variety of possible transformation behaviors while minimizing the number of parameters required. Considering the trade-off between model complexity and fitting accuracy, the 4th-degree polynomial was somewhat holistically selected for modeling the transformation functions.

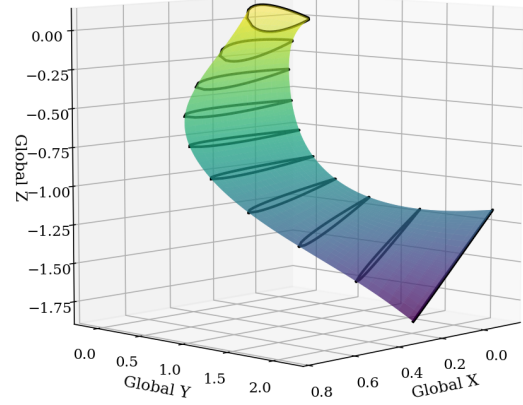
A Non-Uniform Rational B-Spline (NURBS) curve is used to define the blade's centerline due to its flexibility and precision in modeling complex natural shapes.

The generation of the blade centerline begins by dividing the lateral face of the cylindrical hub into Z equal sections. For a section, a local 2D coordinate system is established, which can be seen in Figure 10.

- The positive x -axis ($+x$) extends horizontally across the section.



(a) Example transform polynomials for a blade.



(b) Blade generated as a result.

Figure 9: An example blade generated from given local airfoil transform functions parameterized by s .

- The positive y -axis ($+y$) extends vertically downward towards the bottom of the cylinder.

This coordinate system facilitates the placement and manipulation of control points within each blade's sector. The origin, $(0, 0)$ is placed 1 unit down from the top left of the section. This leaves some padding so that the blade isn't intersecting the top of the cylindrical hub. This 2D coordinate system can be extended to 3D by adding a z -axis, creating a coordinate system denoted L .

- The positive z -axis ($+z$) extends radially out from the cylinder. For any point (x, y) , the scaled z -vector will always normal to the lateral cylinder face.

While extending the z -axis radially creates a non-Euclidean coordinate frame, it creates a natural relation between the z -coordinate and the radial distance of control points, which can help with optimization.

To facilitate the boolean union operation between the blade and the hub, the centerline is inset slightly into the hub's volume. This is achieved through Equation 22 where d is a factor used for the ratio of inset.

$$r_{\text{inset}} = r_{\text{hub}} - \frac{r_{\text{hub}}}{d} \quad (22)$$

This inset ensures that the generated blade intersects with the hub, allowing for a seamless geometric union that results in a single, manifold mesh suitable for simulation and analysis.

A cubic NURBS spline requires control points that define the curve's shape. For each blade, four control points are defined:

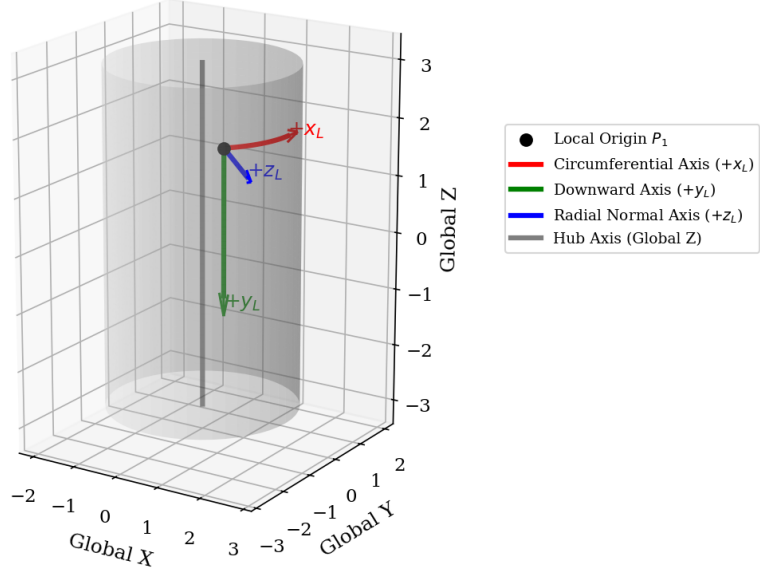


Figure 10: Representation of the coordinate system that is used to define centerline control points.

- $P_1: (0, 0, 0)_L$
- $P_2: (x_2, y_2, z_2)_L$
- $P_3: (x_3, y_3, z_3)_L$
- $P_4: (x_4, y_4, 0)_L$

1. **First Control Point P_1 :** The coordinates of P_1 in the global coordinate system are:

$$P_1 = \left(R_{\text{inset}}, 0, \frac{L_{\text{hub}}}{2} - 1 \right)$$

This is taken such that the local origin for a blade's lateral face section lies on the global x -axis. Thus the coordinate for P_1 in the global cylindrical and Cartesian frame are the same numerically.

2. **Local to Global Coordinate Transformation:** Let C be the global cylindrical frame, represented by coordinates $(r, \theta, z)_C$. Coordinate frame L is simply a remapping of this cylindrical frame, where change in x_L is the same as a change in θ_C , a change in y_L is the same as a change in $-z_C$, and a change in z_L is the same as a change in r_C . Thus, the change of basis matrix becomes:

$$\mathbf{M} = [\mathbf{e}_{x_L} \quad \mathbf{e}_{y_L} \quad \mathbf{e}_{z_L}] = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix} \quad (23)$$

Therefore, for local control point P_i , its relative global cylindrical coordinate is found with:

$$\mathbf{P}_{iC'} = \mathbf{M} \cdot \mathbf{P}_{iL}, \quad (24)$$

$$\mathbf{P}_{iC} = \mathbf{P}_{iC'} + \mathbf{P}_{1C} \quad (25)$$

To get the global coordinates for control point P_i , the offset of origins needs to be considered, so adding the origin vector, which is just $\mathbf{P}_1$, will give the absolute coordinates.

$$\mathbf{P}_{iC} = \mathbf{P}_{iC'} + \mathbf{P}_{1C} \quad (26)$$

With the control points defined, the cubic NURBS spline $C(s)$ is constructed as:

$$C(s) = \frac{\sum_{i=0}^3 N_{i,3}(s)w_i P_i}{\sum_{i=0}^3 N_{i,3}(s)w_i} \quad (27)$$

where:

- $N_{i,3}(s)$ are the cubic B-spline basis functions,
- w_i are the weights associated with each control point. For this case, so as to reduce the parameters needed for optimization, the weights are fixed with $w_i = 1$
- s is the normalized parameter along the spline ($0 \leq s \leq 1$).

The knot vector $\{u_i\}$ for a cubic NURBS with clamped ends is:

$$\{u_i\} = \{0, 0, 0, 0, 1, 1, 1, 1\} \quad (28)$$

The cubic NURBS spline provides the necessary flexibility to define smooth and continuous centerline curves for each blade while maintaining rotational symmetry across the hub. By defining control points in a non-Euclidean coordinate space, where the z -axis is radial with respect to the hub, the resulting centerline ensures that each airfoil section remains perpendicular to the centerline.

The spline fitting process involves adjusting the control points P_1 and P_2 to achieve the desired curvature and alignment of the blade. By manipulating these control points, complex blade geometries can be accurately modeled with a minimal number of parameters.

The Frenet-Serret frame provides an orthonormal basis at each point along the curve, using the tangent vector $T(s)$, the normal vector $N(s)$, and the binormal vector

$B(s)$. The final 3D coordinates of the airfoil are:

$$X_{\text{final}}(s, t) = X_c(s) + X_{\text{scaled}}(s, t)N_x(s) + Y_{\text{scaled}}(s, t)B_x(s) \quad (29)$$

$$Y_{\text{final}}(s, t) = Y_c(s) + X_{\text{scaled}}(s, t)N_y(s) + Y_{\text{scaled}}(s, t)B_y(s) \quad (30)$$

$$Z_{\text{final}}(s, t) = Z_c(s) + X_{\text{scaled}}(s, t)N_z(s) + Y_{\text{scaled}}(s, t)B_z(s) \quad (31)$$

where (X_c, Y_c, Z_c) are the coordinates of the centerline $C(s)$, and (N_x, N_y, N_z) , (B_x, B_y, B_z) are components of $N(s)$ and $B(s)$. The control points and how airfoils are placed along them for a blade can be seen in Figure 6 and Figure 9.

Cell Normalization

Using a linear spacing for the airfoil parameter, t , yields a mesh with uneven cell sizes, which emerges if these equations are used to directly generate a mesh with the parameters t and s . Since

$$\frac{d^2\mathbf{f}}{dt^2} \neq 0 \text{ where } \mathbf{f}(t) = (x(t), y(t))$$

and $\mathbf{f}$ is the vector airfoil function, the derivative of $\mathbf{f}$ changes along the airfoil's surface. This means that an even linearly sampled set for the input will have the output's spacing be inconsistent. This would cause the outputted mesh to have differential cell sizes, with the leading edge, or regions where $|\frac{d\mathbf{f}}{dt}|$ is the largest, having larger cells compared to the trailing edge. This can be corrected with an optional cell normalization step.

Uniformly distributed mesh points along the airfoil perimeter ensure consistent cell sizes in the final mesh, improving CFD and meshing accuracy. The goal is to find some set or vector of parameter values, $\mathbf{t}$, that when plugged into the symbolic equations will create even cell sizes in the mesh. Even sizes of meshes are determined by covering the same arc distance along the airfoil.

The arc length $s(t)$ along the airfoil is:

$$s(t) = \int_{t_0}^t \sqrt{\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2} dt \quad (32)$$

Due to the complexity of the NACA equations, numerical integration is typically employed.

1. Total Arc Length S_{total} :

$$S_{\text{total}} = s(t_{\text{end}}) \quad (33)$$

2. Arc Length Increment Δs :

$$\Delta s = \frac{S_{\text{total}}}{n} \quad (34)$$

where n is the desired number of intervals (mesh resolution).

3. Iterative Calculation of t Values:

Starting from t_0 , solve for t_i such that:

$$s(t_i) - s(t_{i-1}) = \Delta s$$

This is not analytically solvable, so numerical methods are used to approximate the solution.

4. Generate Mapping Function $t(s)$:

A linear interpolation between the integrated data points yields a function $t(s)$, which when inputted the current cumulative arc distance, outputs the t value that would produce the current coordinate point. Evaluating Equation 35 yields the vector $\mathbf{t}$, whose elements are t -values that will produce even mesh spacing.

$$t(\bar{\mathbf{s}}) = \mathbf{t}, \quad \bar{\mathbf{s}} = \{s_i = i \times \Delta s \mid i \in [0, n]\} \quad (35)$$

The blade geometry is duplicated Z times, rotated evenly around the hub's axis:

$$\theta_i = \theta_0 + \frac{2\pi i}{Z} \quad (36)$$

2.3.2 Ye et al. Method

Ye et al. Equations

The equations of toroidal propellers are similar to the equations of screw propellers with the addition of the last two orienting rotations (α and ψ) and propeller reference line variables (l and ϕ) [15]. The toroidal propeller reference line (Equation 37) and the toroidal propeller generator line (Equation 38) are similar to their screw propeller counterparts with the addition of l and ϕ . Equation 39 defines the blade surface, with the inclusion of the rotations α and ψ , which allow for better connectivity between airfoils in the tip region. Because there are two airfoils at each radial location except $r/R = 1$, a new reference line should be chosen. A standardized reference line has not been defined.

$$\begin{cases} x = l \\ r = r \\ \theta = \phi \end{cases} \quad (37)$$

$$\begin{cases} x = l + k + r\theta_s \tan \beta \\ r = r \\ \theta = \phi + \theta_s \end{cases} \quad (38)$$

$$\begin{cases} x = l + k + r\theta_s \tan \beta + (s - \frac{c}{2}) \cos \alpha \sin \beta - y \cos \psi \cos \beta \\ r = r - (s - \frac{c}{2}) \sin \alpha - y \sin \psi \\ \theta = \phi + \theta_s + \frac{1}{r}[(s - \frac{c}{2}) \cos \alpha \cos \beta + y \cos \psi \sin \beta] \end{cases} \quad (39)$$

Point Cloud

The same point cloud approach as the screw propellers is used. Due to the nature of the tip of the toroidal propeller, self-intersections are more common.

Ye et al. Validation

The blade and propeller parameters shown in [15] were attempted first. The paper specified a NACA66-Mod as the airfoil and as no defined geometry was given for this airfoil, the NACA66 $a=0.8$ TMB Mod developed in [13] was used. A copy of this geometry is shown in Table 1. The point cloud is shown in Figure 11. Attempts to build this propeller were unsuccessful due to self-intersections seen in Figure 12.

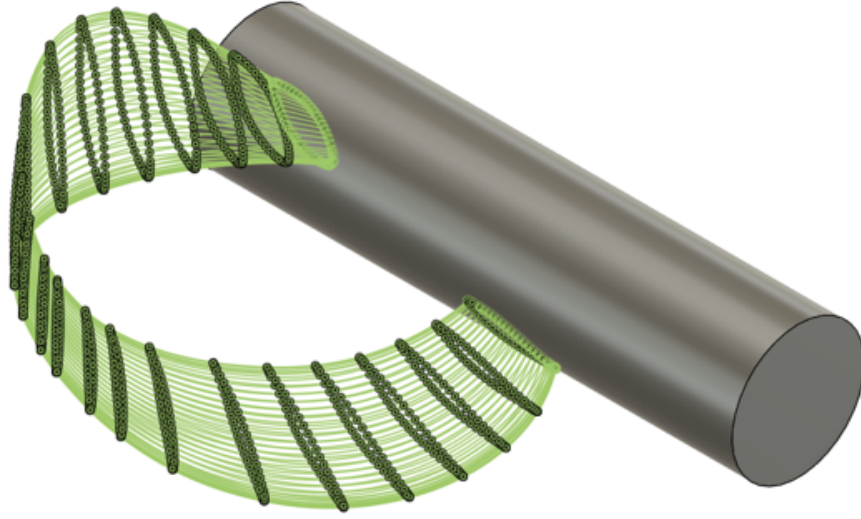


Figure 11: Point cloud with airfoil profiles and guide curves of propeller defined in [15].

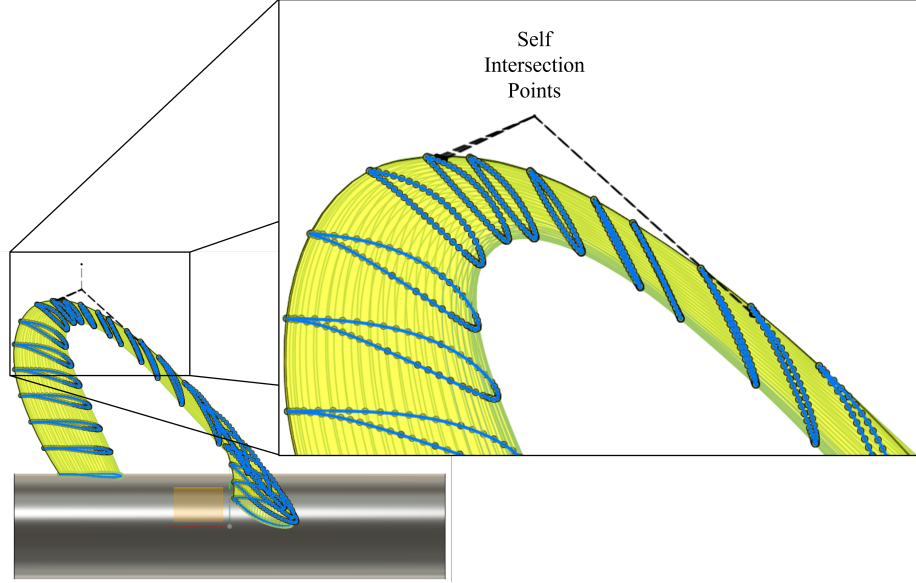


Figure 12: Intersection locations during loft operation of propeller defined in [15].

Multiple attempts at validating these results were unsuccessful. The following modifications were made to fix the self-intersections at the tip and improve the tip shape: 1) Invert the airfoils in the self intersecting region. 2) Modify roll angle to fine tune the shape. From Figure 12 we can determine the radial positions where self-intersections occur ($r/R : [0.950 \rightarrow 1 \rightarrow 0.995]$). Inverting the airfoils involves negating both the thickness and camber. The roll over most of the propeller was also changed. The resulting propeller can be seen in Figure 13 and Figure 14. A reproduction of the complete propeller parameters is seen in Table 3 and Table 4.

Table 2: Select Blade Parameters of [15] and Modifications

r/R	t/c Ye et al.	t/c	f/c Ye et al.	f/c	ψ° Ye et al.	ψ°
0.200	0.3007	0.3007	0.0140	0.0140	0.00	0.00
0.300	0.2433	0.2433	0.0304	0.0304	2.44	2.44
0.400	0.1967	0.1967	0.0440	0.0440	3.38	3.38
0.500	0.1576	0.1576	0.0550	0.0550	4.26	4.26
0.600	0.1253	0.1253	0.0634	0.0634	5.44	5.44
0.700	0.0986	0.0986	0.0689	0.0689	8.31	8.31
0.800	0.0774	0.0774	0.0698	0.0698	14.67	14.67
0.900	0.0608	0.0608	0.0619	0.0619	28.78	15.00
0.950	0.0548	-0.0548	0.0518	-0.0518	42.83	80.00
0.975	0.0526	-0.0526	0.0431	-0.0431	55.45	83.00
0.995	0.0524	-0.0524	0.0298	-0.0298	75.91	86.00
1.000	0.0538	-0.0538	0.0216	-0.0216	90.00	90.00
0.995	0.0583	-0.0583	0.0083	-0.0083	118.06	110.00
0.975	0.0653	0.0653	-0.0035	-0.0035	139.40	205.00
0.950	0.0727	0.0727	-0.0113	-0.0113	151.75	210.00
0.900	0.0867	0.0867	-0.0203	-0.0203	164.02	195.00
0.800	0.1138	0.1138	-0.0268	-0.0268	173.49	192.00
0.700	0.1410	0.1410	-0.0276	-0.0276	176.68	189.00
0.600	0.1686	0.1686	-0.0260	-0.0260	177.66	187.00
0.500	0.1966	0.1966	-0.0230	-0.0230	178.50	185.00
0.400	0.2246	0.2246	-0.0190	-0.0190	179.13	183.00
0.300	0.2526	0.2526	-0.0147	-0.0147	179.62	181.00
0.200	0.2807	0.2807	-0.0100	-0.0100	180.00	180.00

Table 3: Blade Parameters of [15] with Modifications

l/L	r/R	c/D	P/D	t/c	f/c	θ_S°	k/D	ϕ°	ψ°	α°
0.0000	0.200	0.1210	1.3120	0.3007	0.0140	0.00	-0.0620	-27.56	0.00	0.00
0.0624	0.300	0.1431	1.3311	0.2433	0.0304	-0.89	-0.0822	-25.52	2.44	0.12
0.1203	0.400	0.1600	1.3365	0.1967	0.0440	-1.10	-0.0933	-23.44	3.38	0.62
0.1773	0.500	0.1723	1.3206	0.1576	0.0550	-0.90	-0.0968	-21.26	4.26	1.44
0.2343	0.600	0.1797	1.2807	0.1253	0.0634	-0.36	-0.0928	-18.93	5.44	2.61
0.2937	0.700	0.1806	1.1980	0.0986	0.0689	0.43	-0.0807	-16.33	8.31	4.37
0.3590	0.800	0.1683	1.0784	0.0774	0.0698	1.46	-0.0587	-13.26	14.67	6.89
0.4377	0.900	0.1423	0.9317	0.0608	0.0619	2.83	-0.0255	-9.33	15.00	9.39
0.4889	0.950	0.1228	0.8631	-0.0548	-0.0518	3.78	-0.0022	-6.55	80.00	10.45
0.5250	0.975	0.1120	0.8300	-0.0526	-0.0431	4.45	0.0140	-4.56	83.00	10.98
0.5729	0.995	0.1027	0.8065	-0.0524	-0.0298	5.34	0.0347	-1.86	86.00	11.41
0.6000	1.000	0.1010	0.8060	-0.0538	-0.0216	5.87	0.0458	-0.24	90.00	11.51
0.6438	0.995	0.1065	0.8528	-0.0583	-0.0083	6.69	0.0615	2.45	110.00	11.28
0.6842	0.975	0.1190	0.9512	0.0653	-0.0035	7.44	0.0732	4.99	205.00	10.52
0.7146	0.950	0.1311	1.0400	0.0727	-0.0113	7.99	0.0789	6.95	210.00	8.97
0.7574	0.900	0.1514	1.1829	0.0867	-0.0203	8.75	0.0819	9.78	195.00	5.54
0.8150	0.800	0.1763	1.3959	0.1138	-0.0268	9.72	0.0723	13.77	192.00	2.33
0.8582	0.700	0.1869	1.5433	0.1410	-0.0276	10.40	0.0537	16.83	189.00	1.44
0.8941	0.600	0.1862	1.6367	0.1686	-0.0260	10.94	0.0298	19.44	187.00	0.98
0.9249	0.500	0.1770	1.6686	0.1966	-0.0230	11.39	0.0024	21.74	185.00	0.63
0.9523	0.400	0.1626	1.6636	0.2246	-0.0190	11.77	-0.0272	23.82	183.00	0.38
0.9771	0.300	0.1447	1.6524	0.2526	-0.0147	12.11	-0.0584	25.74	181.00	0.17
1.0000	0.200	0.1240	1.6310	0.2807	-0.0100	12.41	-0.0910	27.56	180.00	0.00

Table 4: Blade Parameters of [15]

Parameter	Value
Diameter (D) [m]	0.3048
Hub Radius (r_h/R)	0.20
Wheelbase (L/D)	0.30
Number of Blades (Z)	3
Airfoil	NACA66 $a = 0.8$ TMB Mod

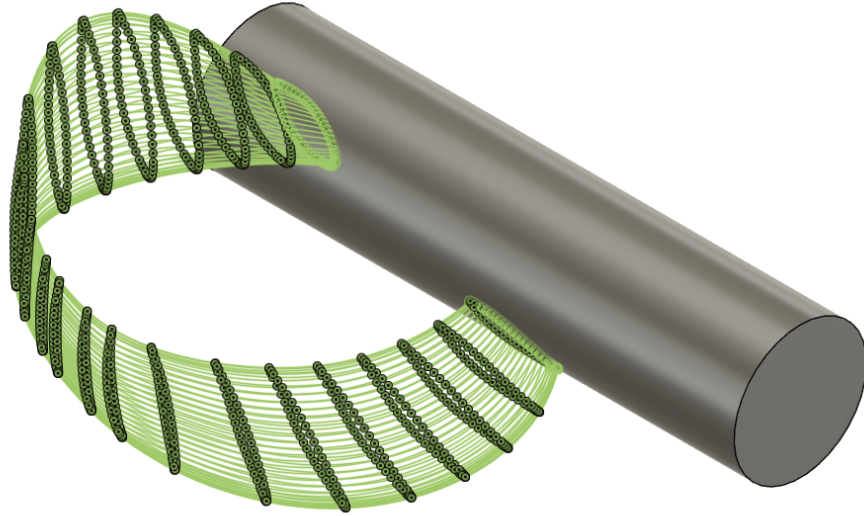


Figure 13: Point cloud with airfoil profiles and guide curves of modified [15] propeller as per Table 2.

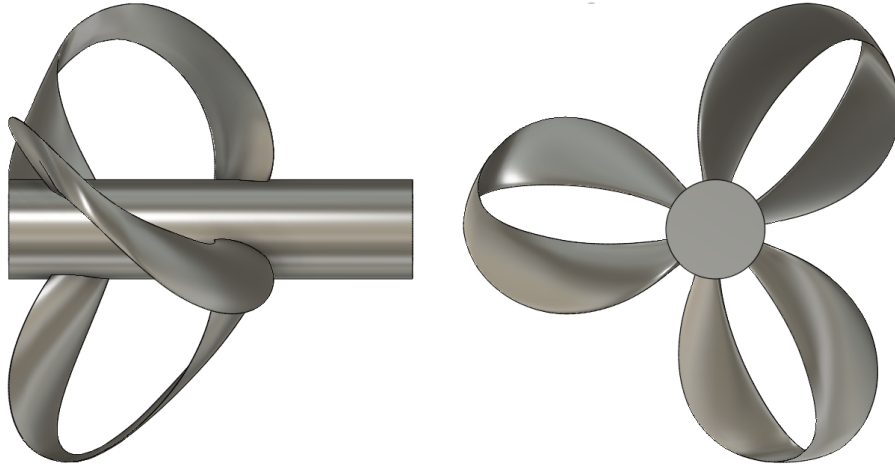


Figure 14: Successful geometry creation of modified [15] propeller as per Table 2.

2.4 4119 and BAR Matching

The toroidal propellers are compared to the Carderock 4119 screw propeller in this project, with matched EAR.

2.4.1 Screw Propeller 4119 Parameters

The 4119 was generated with the blade parameters shown in Table 5 and propeller parameters shown in Table 6.

Table 5: 4119 Blade Parameters. Adapted from [24].

r/R	c/D	t/c	f/c	P/D	k/D	θ_s°
0.20	0.3200	0.2055	0.0143	1.1050	0.0000	0.00
0.30	0.3625	0.1552	0.0232	1.1020	0.0000	0.00
0.40	0.4048	0.1180	0.0230	1.0980	0.0000	0.00
0.50	0.4392	0.0902	0.0218	1.0930	0.0000	0.00
0.60	0.4610	0.0696	0.0207	1.0880	0.0000	0.00
0.70	0.4622	0.0542	0.0200	1.0840	0.0000	0.00
0.80	0.4347	0.0421	0.0197	1.0810	0.0000	0.00
0.90	0.3613	0.0332	0.0182	1.0790	0.0000	0.00
0.95	0.2775	0.0323	0.0163	1.0770	0.0000	0.00
1.00	0.0100	0.0316	0.0118	1.0750	0.0000	0.00

Table 6: 4119 Propeller Parameters. Partially adapted from [24].

Parameter	Value
Diameter (D) [m]	0.3048
Hub Radius (r_h/R)	0.20
Number of Blades (Z)	3
Airfoil	NACA66 $a = 0.8$ TMB Mod

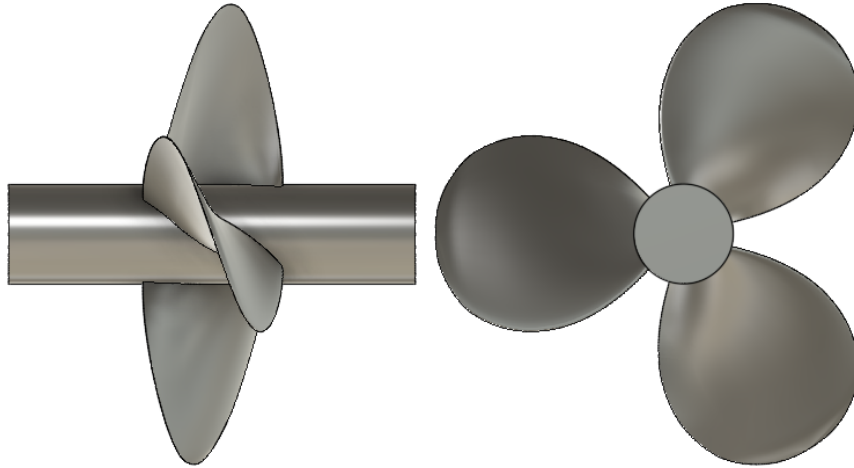


Figure 15: Successful geometry creation of the 4119 propeller as per Table 5 and Table 6.

2.4.2 Blade Area Ratio Matching

While EAR matching does not directly match surface area, it is quite similar. An increase in surface area correlates to an increase in thrust. Holding EAR constant

across all of the propellers constrains the chord length, c , and allows a baseline to be created over which other parameters can alter efficiency.

The 4119 is said to have an A_E/A_0 of 0.6 [24]. This was confirmed by applying the appropriate definition and connecting each expanded airfoil with either a fit point spline (resulting in an EAR of 0.607) or lines (0.599). We will be using the trapezoidal integration method for simplicity as seen in Equation 41.

$$A_E = Z \int_{r_h}^R c(r) dr \quad (40)$$

$$A_E/D = Z \sum_{i=r_h}^R \frac{1}{4} \left(\frac{b_i}{D} + \frac{b_{i+1}}{D} \right) \left| \left(\frac{r_i}{R} + \frac{r_{i+1}}{R} \right) \right| \quad (41)$$

$$A_0 = \frac{\pi D^2}{4} \quad (42)$$

The initial EAR matched propeller (TORUS 0000) was created by combining elements of the 4119 propeller, the toroidal propeller shown in [15], and the revisions done to the latter propeller. The following parameters were directly used from [15]: l/L , ϕ , and α . Since 4119 has no skew and no rake, θ_s and k/D were left to be zero. The following parameters were directly used from [24] with the exception of the r/R values of 0.975 and 0.995 where interpolation was used: P/D , t/c , and f/c . Some thickness and camber values needed to be negated as described earlier. Similar ψ values as those used in during Section 2.3.2 were used.

Chord length is the only parameter affected in EAR matching. The c/d values in [15] were smoothed with a piecewise approximation. The $r/R = 1$ value is the average of the front and rear surface functions. EAR was calculated and c/d was uniformly scaled to match the desired EAR of $A_E/A_0 = 0.6$.

$$c/D\left(\frac{r}{R}\right) = \begin{cases} a_1 \left(\frac{r}{R}\right)^3 + a_2 \left(\frac{r}{R}\right)^2 + a_3 \left(\frac{r}{R}\right) + a_4, & \frac{r}{R} = [0.2 \rightarrow 1) \\ \frac{1}{2} \left[\lim_{\frac{r}{R} \rightarrow 1^-} c/D\left(\frac{r}{R}\right) + \lim_{\frac{r}{R} \rightarrow 1^+} c/D\left(\frac{r}{R}\right) \right], & \frac{r}{R} = 1 \\ a_1 \left(\frac{r}{R}\right)^3 + a_2 \left(\frac{r}{R}\right)^2 + a_3 \left(\frac{r}{R}\right) + a_4, & \frac{r}{R} = (1 \rightarrow 0.2] \end{cases} \quad (43)$$

Table 7: c/D smoothing approximation variables.

	Front	Rear
a_1	-0.390659	-0.528268
a_2	0.266898	0.496802
a_3	0.137403	0.032659
a_4	0.086842	0.102933

Table 8: TORUS 0000 Blade Parameters

l/L	r/R	c/D	P/D	t/c	f/c	θ_s°	k/D	ϕ°	ψ°	α°
0.0000	0.200	0.1497	1.105	0.2055	0.0143	0.00	0.0000	-27.56	0.00	0.00
0.0624	0.300	0.1739	1.102	0.1552	0.0232	0.00	0.0000	-25.52	2.44	0.12
0.1203	0.400	0.1960	1.098	0.1180	0.0230	0.00	0.0000	-23.44	3.38	0.62
0.1773	0.500	0.2131	1.093	0.0902	0.0218	0.00	0.0000	-21.26	4.26	1.44
0.2343	0.600	0.2224	1.088	0.0696	0.0207	0.00	0.0000	-18.93	5.44	2.61
0.2937	0.700	0.2209	1.084	0.0542	0.0200	0.00	0.0000	-16.33	8.31	4.37
0.3590	0.800	0.2059	1.081	0.0421	0.0197	0.00	0.0000	-13.26	14.67	6.89
0.4377	0.900	0.1743	1.079	0.0332	0.0182	0.00	0.0000	-9.33	-15.00	9.39
0.4889	0.950	0.1515	1.077	0.0323	0.0163	0.00	0.0000	-6.55	-20.00	10.45
0.5250	0.975	0.1382	1.076	0.0320	0.0148	0.00	0.0000	-4.56	-25.00	10.98
0.5729	0.995	0.1265	1.076	0.0318	0.0133	0.00	0.0000	-1.86	-10.00	11.41
0.6000	1.000	0.1257	1.075	-0.0316	-0.0118	0.00	0.0000	-0.24	130.00	11.51
0.6438	0.995	0.1313	1.076	-0.0318	-0.0133	0.00	0.0000	2.45	100.00	11.28
0.6842	0.975	0.1443	1.076	0.0320	-0.0148	0.00	0.0000	4.99	230.00	10.52
0.7146	0.950	0.1590	1.077	0.0323	-0.0163	0.00	0.0000	6.95	210.00	8.97
0.7574	0.900	0.1838	1.079	0.0332	-0.0182	0.00	0.0000	9.78	195.00	5.54
0.8150	0.800	0.2169	1.081	0.0421	-0.0197	0.00	0.0000	13.77	192.00	2.33
0.8582	0.700	0.2310	1.084	0.0542	-0.0200	0.00	0.0000	16.83	189.00	1.44
0.8941	0.600	0.2301	1.088	0.0696	-0.0207	0.00	0.0000	19.44	187.00	0.98
0.9249	0.500	0.2180	1.093	0.0902	-0.0218	0.00	0.0000	21.74	185.00	0.63
0.9523	0.400	0.1986	1.098	0.1180	-0.0230	0.00	0.0000	23.82	183.00	0.38
0.9771	0.300	0.1759	1.102	0.1552	-0.0232	0.00	0.0000	25.74	181.00	0.17
1.0000	0.200	0.1537	1.105	0.2055	-0.0143	0.00	0.0000	27.56	180.00	0.00

Table 9: TORUS 0000 Propeller Parameters

Parameter	Value
Diameter (D) [m]	0.3048
Hub Radius (r_h/R)	0.20
Wheelbase (L/D)	0.30
Number of Blades (Z)	3
Airfoil	NACA66 $a = 0.8$ TMB Mod

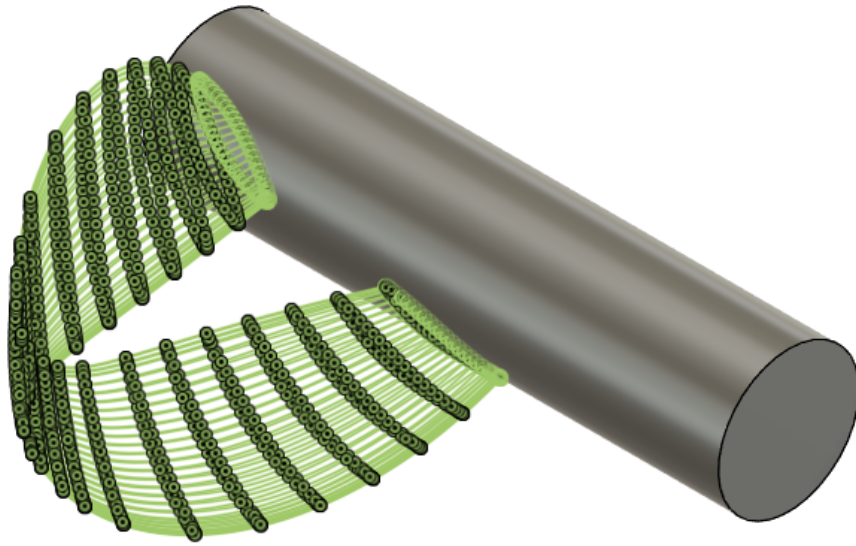


Figure 16: Point cloud with airfoil profiles and guide curves of TORUS 0000 propeller as per Table 8 and Table 9.

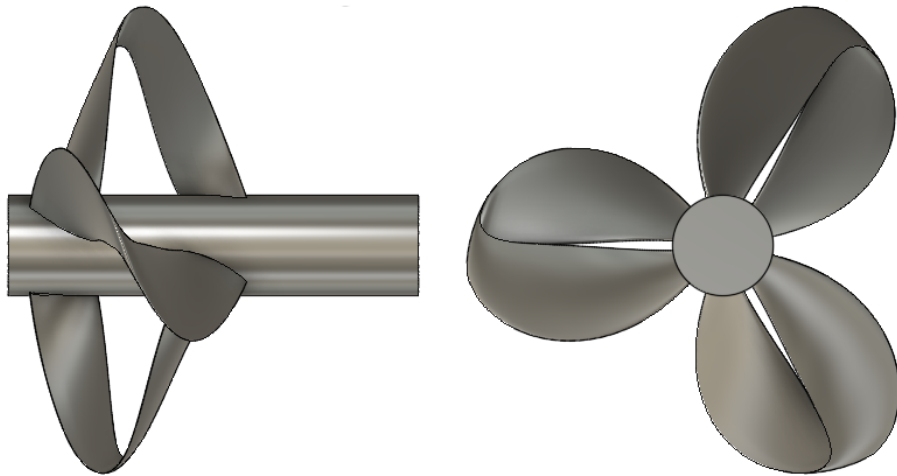


Figure 17: Successful geometry creation of TORUS 0000 propeller as per Table 8 and Table 9.

3 Analysis Methods

3.1 CFD

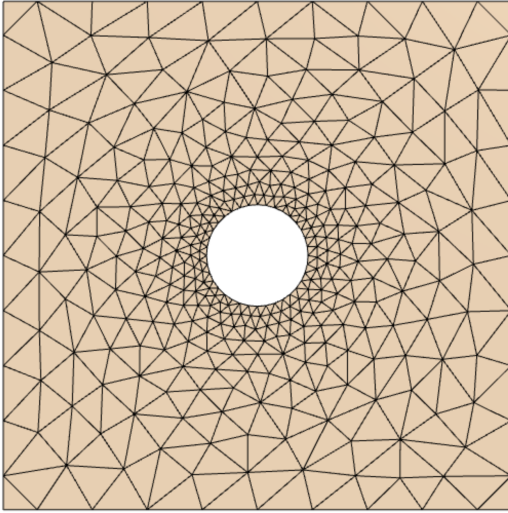
In the context of marine engineering, traditional practices are reliant on empirical data and trial-and-error approaches. However, this poses severe limitations in adaptability and precision with increasing complexity of marine engineering designs. Simulation-based design optimization (SBDO) enables nuanced exploration of design possibilities. In the specific context of marine propulsion systems, SBDO allows designers to deal with the complexity of components like engines, propellers, shafts and rudders, and facilitates the optimization of these components for maximum efficiency and reduced fuel consumption. The approach with SBDO combines traditional CFD simulation with optimization algorithms, enabling a more systematic, flexible and precise methodology for studying designs. Serani, et. al [25] identified 277 SBDO studies in marine engineering up to 2022. The number of publications showed a steady increase starting in 2009, with a substantial spike in 2015, and the earliest SBDO paper being from 1994. Of the papers included in this study, hull hydrodynamics was the largest application category, using CFD solvers to evaluate drag, resistance, propulsion efficiency and flow behavior.

Due to the increased flexibility for design and performance analysis, a wide range of CFD packages exist, both commercial and open-source. Examples include STAR-CCM+, ANSYS Fluent, ANSYS CFX, OpenFOAM (open-source), and SU2 (open-source). These packages differ in what type of fluid flow problem they are tailored toward, covering a wide range of use-cases from academic research to industrial design and optimization. However, the underlying procedures involved in solving the fluid flow problems are fundamentally similar. All CFD packages obtain solutions to the governing equations of fluid flow, these being continuity, momentum, and energy. These solutions are often numerical approximations obtained using the finite-volume method whereby the governing equations are integrated over discrete control volumes and converted into algebraic equations by balancing convective, diffusive, and source fluxes. Pressure-velocity coupling algorithms such as SIMPLE and PISO are then used to enforce mass conservation. Boundary conditions (inlet, outlet, wall, and symmetry) must be defined, and emphasis must be put on the correct implementation of turbulence quantities and near-wall treatment. The discretized equations rely on spatial schemes (upwind, central differencing, QUICK) and temporal schemes (explicit, implicit, Crank-Nicolson), all of which require careful attention to grid generation, numerical stability, residual reduction, and mesh-independence testing. These guidelines form the basis for reliable and physically consistent CFD simulations in engineering applications [26].

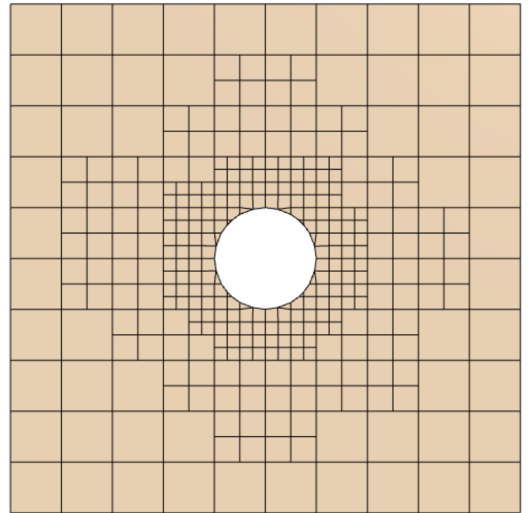
The general workflow in CFD can be broken down into four main steps: modeling, meshing, solving, and post-processing/analysis. Modeling involves creating the geometry that is to be studied (2D or 3D). As CFD is a tool for studying fluid flow, a bounding box is then created to serve as the flow domain, from which this geometry will be subtracted [27]. As most flow fields are naturally turbulent resulting in computationally expensive fluctuations, CFD codes incorporate turbulence models.

These turbulence models are sets of equations that apply to general turbulent motion without resolving every tiny eddy [28].

The discretization of the model is known as meshing. As explained above, this step involves simplifying the fluid flow problem of the complex geometry into a series of smaller, simplified finite-volume elements (cells) for which the fluid flow problem is solved. Key parameters in this process are the resolution and type of mesh [29]. Mesh resolution is dependent on the number of elements that a geometry is broken down into, with higher resolution meshes offering increased accuracy in results at the cost of computation time and resources [27]. Critical regions require greater mesh densities. Mesh types can be structured or unstructured. The latter typically conform easily to complex geometries [29], formed with triangular (2D) elements or hexahedrons (3D). Conversely, structured meshes are divided into ordered, quadrilateral cells, resulting in a simpler grid and reduced computation requirements. Figure 18 shows examples of unstructured (a) and structured (b) meshes for a simple geometry developed in ANSYS. Once an adequate mesh has been generated, boundary and initial conditions must be defined. This can involve defining fluid properties, and inlet, outlet and wall conditions at the respective surfaces.



(a) An unstructured mesh composed entirely of triangles.



(b) A structured mesh composed entirely of quadrilaterals with sizing constraints applied on the edges of the square.

Figure 18: Meshes created in STAR-CCM+ to illustrate types of meshes.

Once an appropriate mesh has been developed, the CFD packages then perform the solving step, iteratively solving conservation of mass and momentum equations. Solutions are obtained by taking linear approximations of these non-linear conservation equations. Often this process introduces discretization and linearization errors. The solution to the mass conservation equations are subject to discretization errors. This is reduced by refining the mesh. Solutions to the momentum conservation equation are subject to linearization errors, which are reduced through iteration by the

Newton-Raphson method. Linearization errors are reduced until a specified tolerance is met, upon which the iterative solution concludes and the discretization errors are minimal [27]. It is important to note here that not all simulations will reach the tolerance conditions imposed. Solutions may diverge if issues with the mesh resolution, time-step size, or geometry are present. In extreme cases, the solver may not even be able to initiate, prompting the user to re-evaluate the model, or solvers will stop at a user-specified iteration number if convergence has not occurred. However, for smaller errors, it is possible for solutions to converge but the results to be inaccurate. In these cases, it is essential to scrutinize the outputs to verify that the results are consistent with expected behavior.

The last step in the CFD workflow is to perform analysis and post-processing of the solver outputs. This is often dependent on the software packages, though most programs offer built-in functions to plot parameters like thrust, drag and pressure coefficients in graph, vector or nodal form. See Figures 19 and 20 for examples.

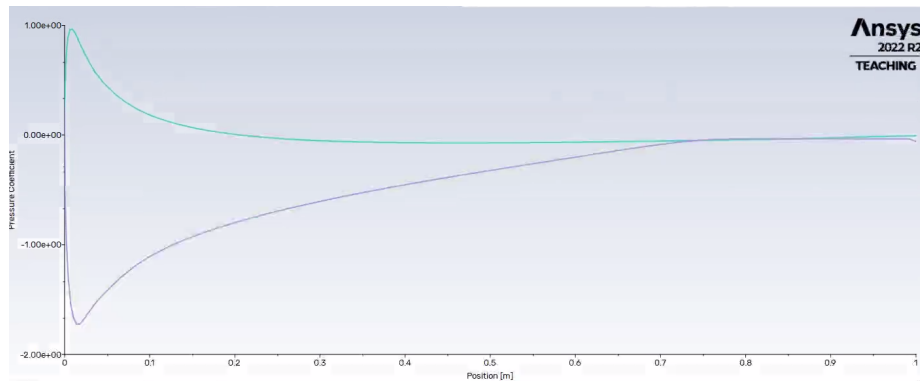


Figure 19: Pressure coefficient graph of NACA0012 airfoil.

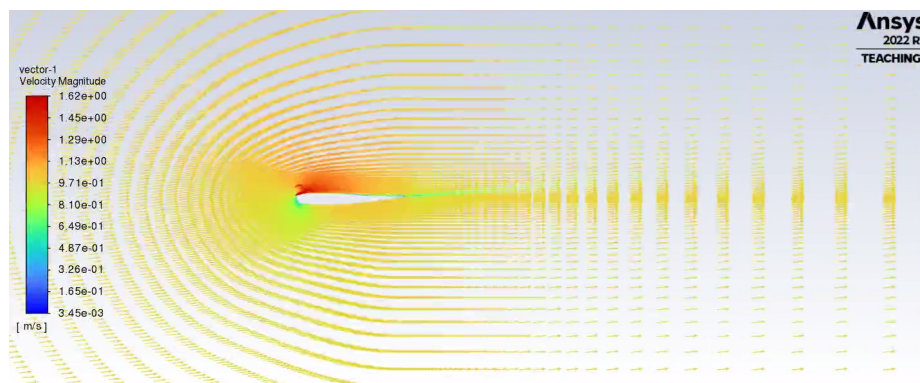


Figure 20: Velocity magnitude vector graph.

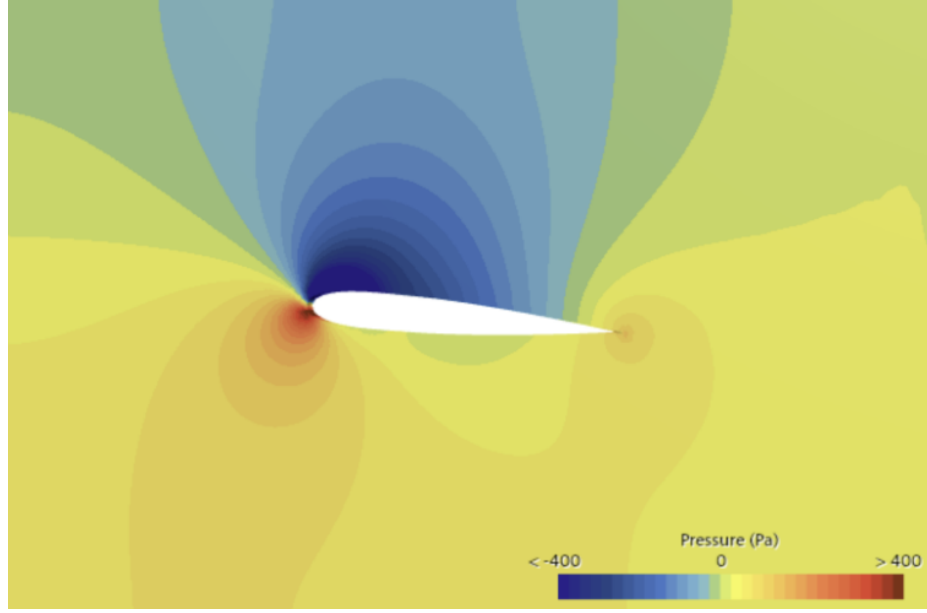


Figure 21: Pressure distribution along a NACA0012 with 5° angle of attack.

3.2 Propeller Simulations

In order to analyze the open water performance of the generated propellers, a moving reference frame (MRF) CFD simulation was created in STAR-CCM+ 2406. The simulation was used to determine K_T , K_Q , and η_0 . To verify the results of the simulation, it was built around the 4119 propeller, which had its full geometric definition [24], operation specifications, and open water test data published [30].

3.2.1 Screw Propeller

Fluid Domain

The region used for this simulation was one third of a cylinder with periodic boundaries to act as the full propeller. The propeller was placed into the domain so that the center of its blade was aligned with the center of the 120° slice. The domain was then divided into three regions: rotating, static, and refinement regions. The shaft of the propeller was extended to the inlet and outlet to avoid wake caused by blunt faces.

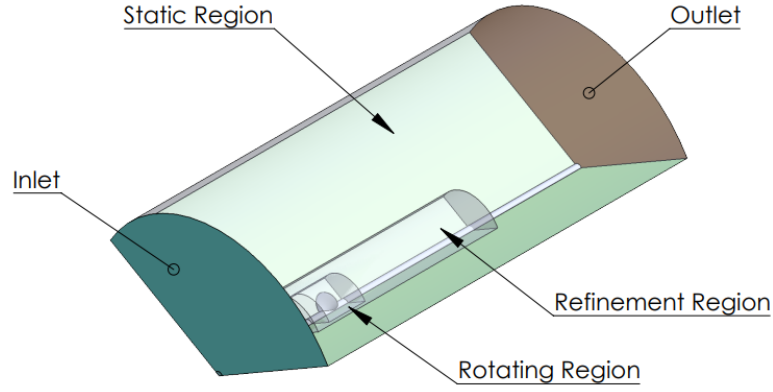


Figure 22: Domain regions with inlet and outlet locations.

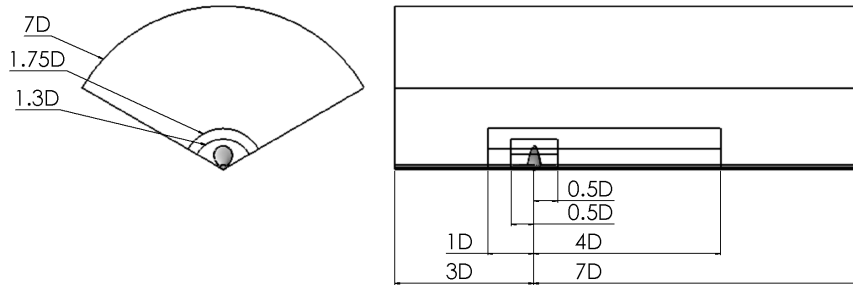


Figure 23: Dimensions of fluid regions.

Once the full shaft was inserted into the domain, the propeller and shaft were subtracted from the rotating and static regions. The surface created during this operation was treated as a wall. The front face of the static region was a velocity inlet with a variable velocity depending on the advance ratio, and the back face of the static region was set to be a pressure outlet set to 0 Pa. The outer cylindrical surface was defined as a symmetry plane and the two slice faces were set to be periodic interfaces.

Mesh

After the boundary conditions were set up, STAR-CCM+'s automated meshing operations were used for the rotating and static regions. Both regions used STAR-CCM+'s surface remesher and trimmed cell mesher.

However, the rotating region featured a denser mesh and prism layers close to accurately capture the boundary layer and create a Y^+ of less than 1 for most of the propeller (Figure 25). The refinement region was used as a volumetric mesh control to create a denser area in the static region where the outflow of the propeller would be. The final mesh can be seen in Figure 30.

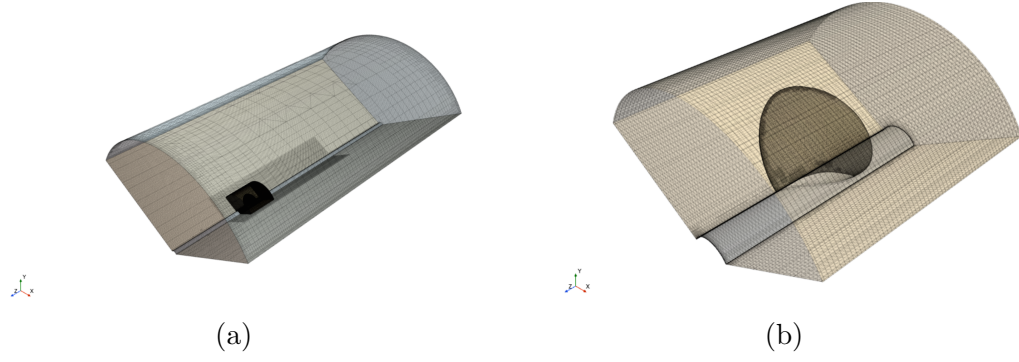


Figure 24: Mesh of full domain and rotating region.

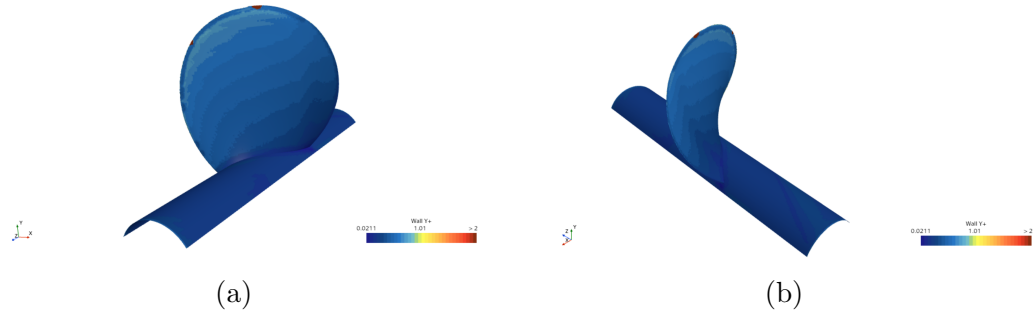


Figure 25: Y^+ values along the front surface and back surface of a 4119.

Screw Propeller Simulation Results and Validation

To verify the CFD's performance across the entire open water test, the advance ratio was varied from 0 to 1.3. Rotational speed was kept constant at 10.435 rps counterclockwise and inlet velocity magnitude was varied according to the advance ratio. For each advance ratio, the simulation ran through 200 outer iterations, and the final values for K_T , $10K_Q$, and η_0 were taken. The rotational speed was kept constant at the design speed, and the velocity was allowed to change with the advance ratio.

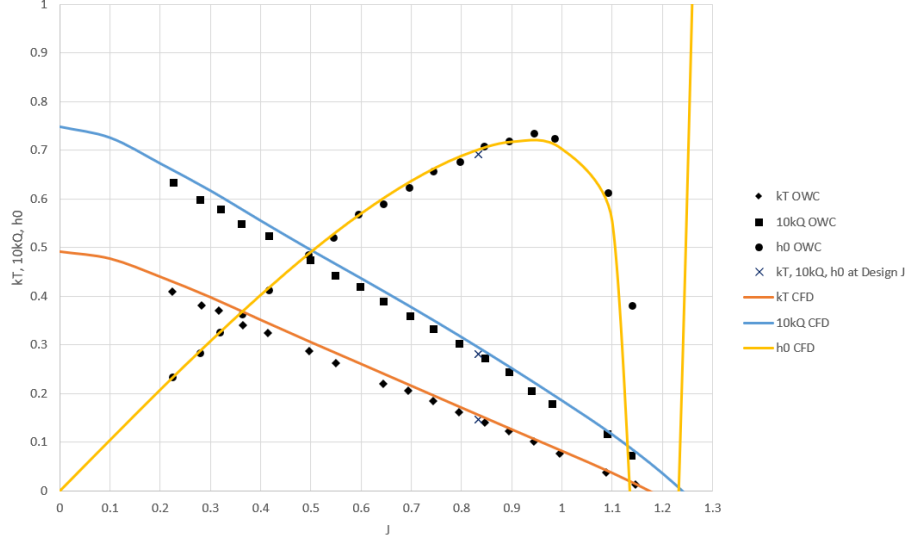


Figure 26: CFD simulation results compared to open-water performance curves [30].

The simulation's results for efficiency were within 2% error for advance ratios between 0 and 1.1, which establishes that the simulation is able to handle the various run conditions expected throughout the optimization process.

Mesh and Iteration Dependence Tests

In order to determine whether the results were influenced by the size of the mesh or the number of iterations, half and double density meshes were created. These, along with the original, were run through the entire advance ratio sweep with 500 iterations at each ratio. The results for K_T , K_Q , and η_0 were taken at 100, 200, and 500 iterations for each as well.

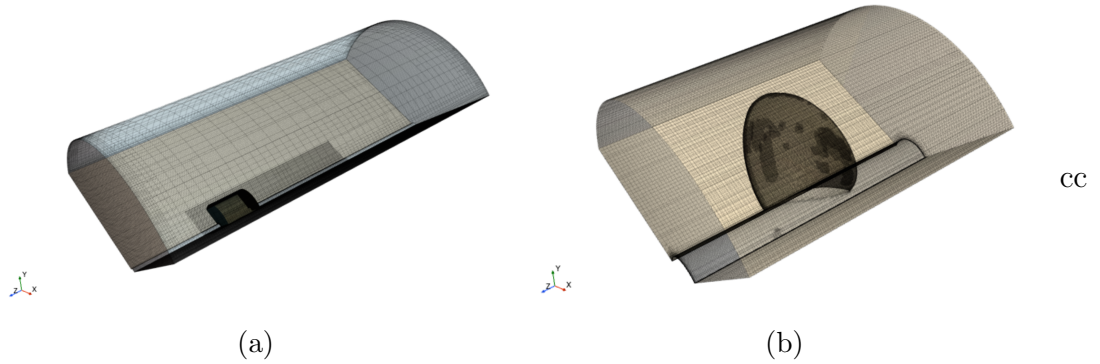


Figure 27: Double density mesh of full domain (a) and rotating region (b).

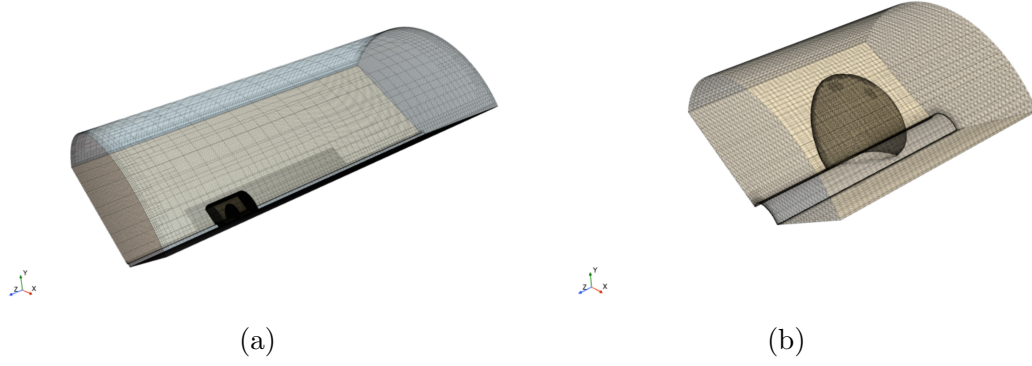
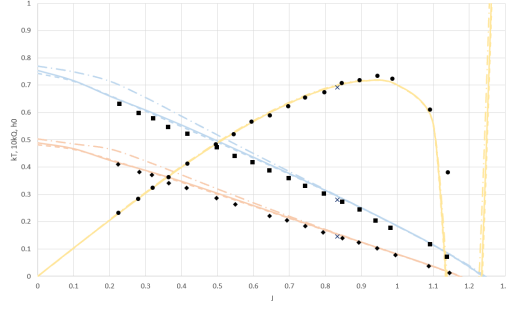


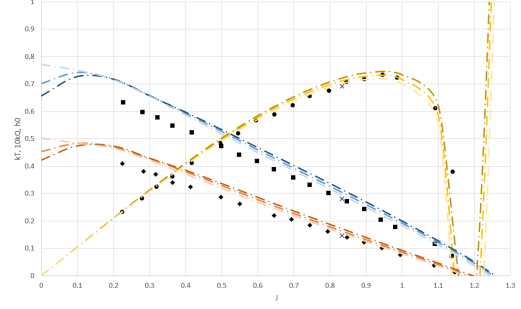
Figure 28: Half density mesh of full domain (a) and rotating region (b).

Table 10: Number of cells in each mesh density.

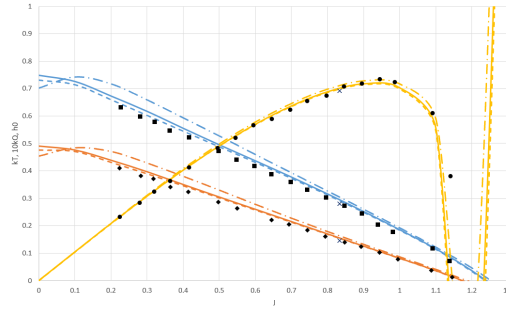
	Double Density	Base Density	Half Density	Half Density v2
Rotating Region	1794923	972938	612825	773578
Static Region	2042629	1156180	602983	602983
Total	3837552	2129118	1215808	1376561



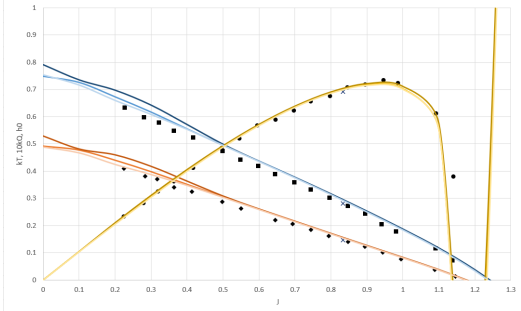
(a) Iterations at Half Density.



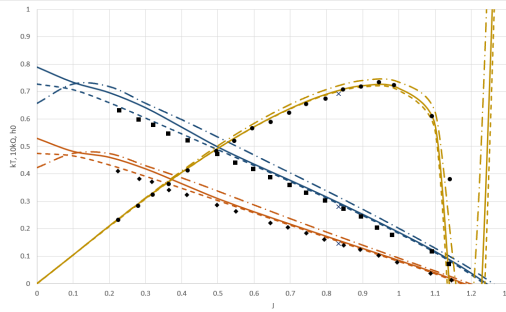
(b) Densities at 100 Iterations.



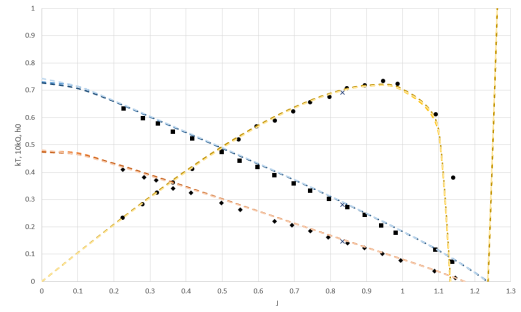
(c) Iterations at Base Density.



(d) Densities at 200 Iterations.



(e) Iterations at Double Density.



(f) Densities at 500 Iterations.

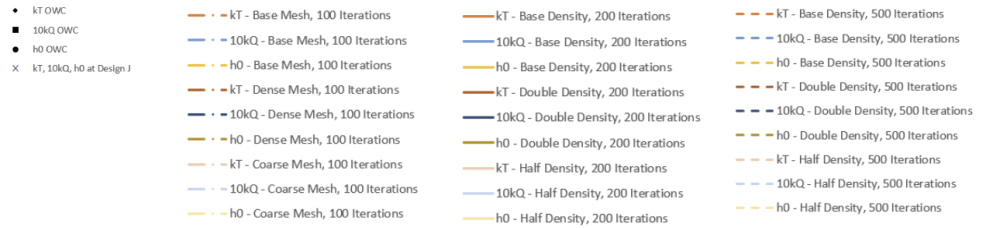


Figure 29: Comparison of results from mesh and iteration dependence study for screw propellers.

From the graphs shown in Figure 29, it can be seen that for 500 iterations, the results barely changed depending on the densities, but for 200 iterations, only the base

and dense mesh appeared the same, and for 100, none were the same. This means that at 100 iterations, none of the results were independent of the mesh, so this was not used. Figure 29 also shows that for both the base and double densities, 200 and 500 iterations had very similar results. Since 200 iterations at the base density had the fastest run time (13 minutes per advance ratio), it was deemed to be the better than the others. However, one more attempt at time reduction was made by creating a new mesh that kept the half density resolution in the static region but added more cells to the rotating region.

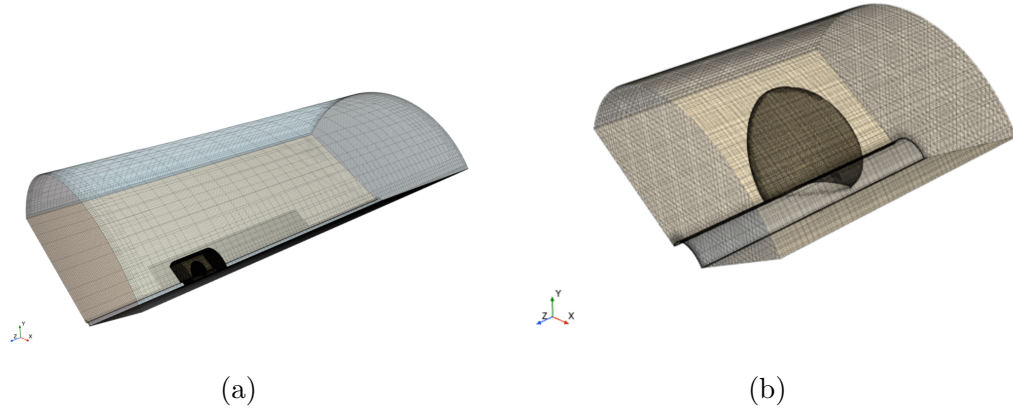


Figure 30: Half density mesh v2 of full domain (a) and rotating region (b).

After creating and running the final mesh, the percent differences between the base and the three others were calculated. For advance ratios between 0.4 and 1.1, it was found that all four meshes had a relative error of 0.05% to the base mesh (Figure 31).

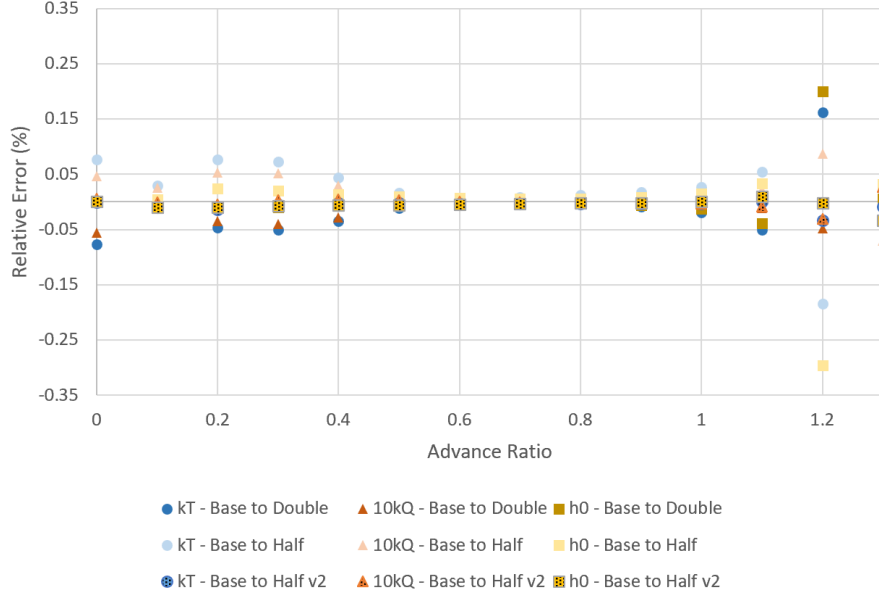


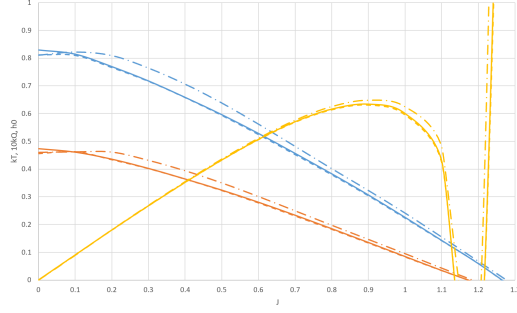
Figure 31: Relative error between base mesh and the three other versions.

3.2.2 Toroidal Propeller

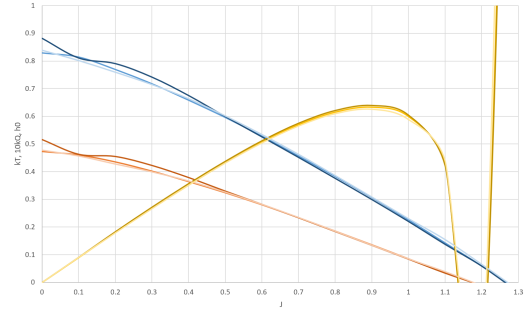
Toroidal propellers can be simulated the same way as screw propellers. However, the new propellers in this study do not have open water curves to validate the results. Since the simulation for the screw propellers was mesh independent, the toroidal propeller should be able to be substituted without causing it to lose accuracy. For the design of the toroidal mesh, TORUS 0000 was used.

Mesh and Iteration Dependence Tests

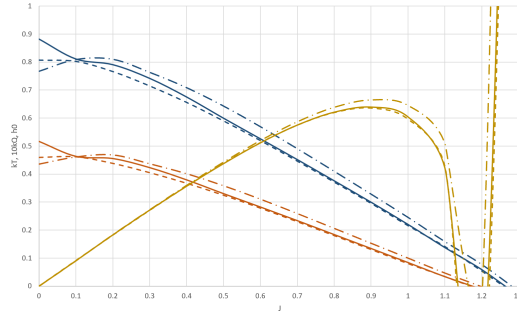
Because the toroidal propellers have a more complex geometry, the mesh independence study was rerun with all four meshes. This test showed similar results to the screw propeller tests. Like the screw propeller, 200 and 500 iterations had nearly identical results for the base and double density meshes between advance ratios of 0.6 and 1.1.



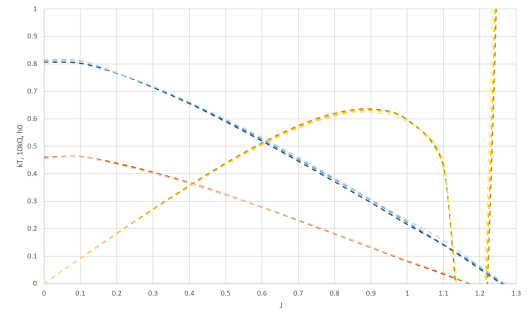
(a) Iterations at Base Density.



(b) Densities at 200 Iterations.



(c) Iterations at Double Density.



(d) Densities at 500 Iterations.



Figure 32: Comparison of results from mesh and iteration dependence study for toroidal propellers.

The second version of the half density was also used for the toroidal propeller, and the percent differences were calculated. The different meshes had a greater difference for the toroidal propellers than for the screw propellers, but the difference was still within tolerance.

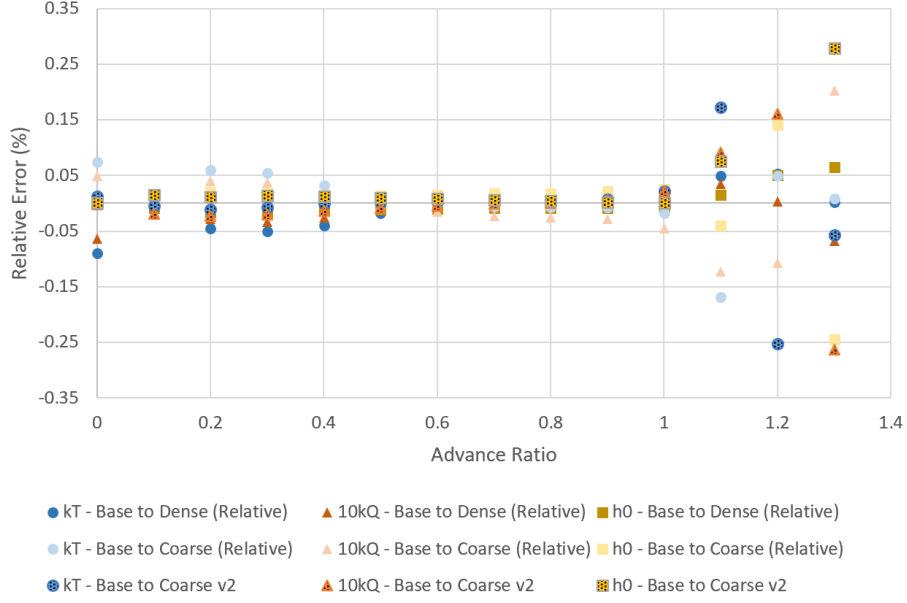


Figure 33: Relative error between base mesh and the three other versions.

Although the results for K_T , $10K_Q$, and η_0 were within tolerance for all meshes, the coarser meshes did not have enough resolution between the blades of the toroidal propeller for flow visualization. Therefore, it was decided to sacrifice time and add an extra refinement region around the blades. This new refined mesh will be used for the final simulations.

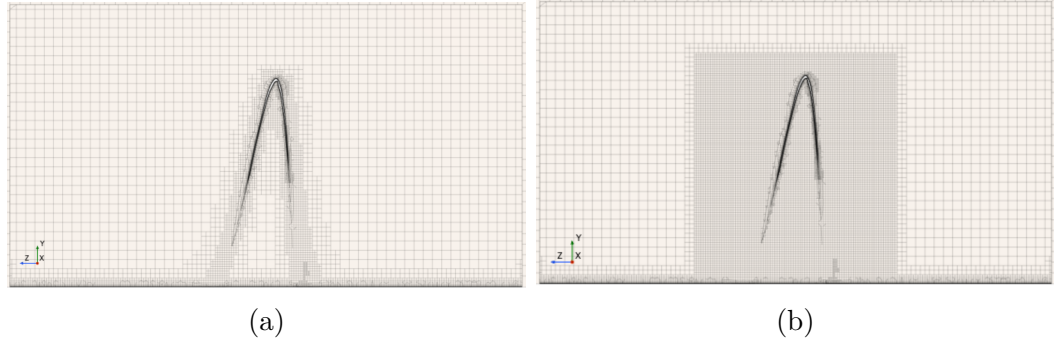


Figure 34: yz-plane section cut of toroidal propeller simulation's rotating region showing resolutions of half density version two (a) and new refined region (b).

3.2.3 Final CFD Simulation

Up to this point, the simulations have been running through a sweep of advance ratios. However, due to time considerations, it was decided to run each propeller at the 4119's design specifications, which would drop time per propeller from 3.5 hours to 15 minutes. Denny [30] specified that the 4119 was designed to run at an advance ratio of $J = 0.833$ and that the corresponding Reynolds number at 0.7R be roughly

1×10^6 [30]. From these two values, the inlet velocity and the rotational speed were calculated (Table 11).

Table 11: Run conditions for the 4119.

Advance Ratio	0.833
Reynolds Number _{0.7D}	1×10^6
Inlet Velocity	2.65 m/s
Flow Direction	-Z
Rotational speed	10.435 rps
Rotation Direction	CCW

As demonstrated by Figure 35, K_T , $10K_Q$, and η_0 reached a convergence by the final iteration of the simulation.

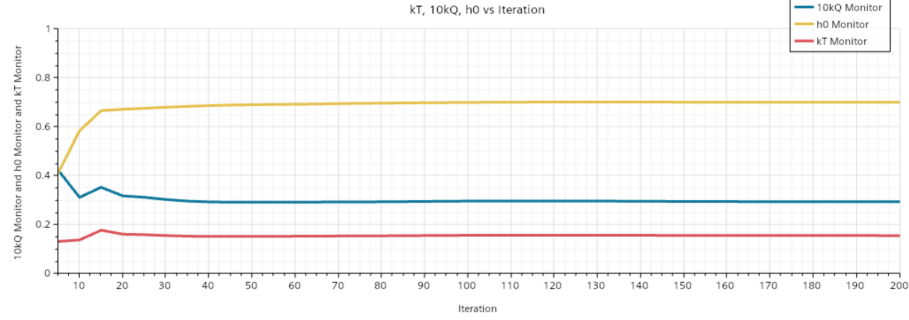


Figure 35: CFD results of the 4119 propeller run at design conditions.

Additionally, the single advance ratio simulation allowed flow visualization for each propeller to be created. For example, two visualizations considered were the velocity in the z-direction and the pressures at different radii of the propeller blades.



Figure 36: Velocity in the z-direction shown on the yz-plane.

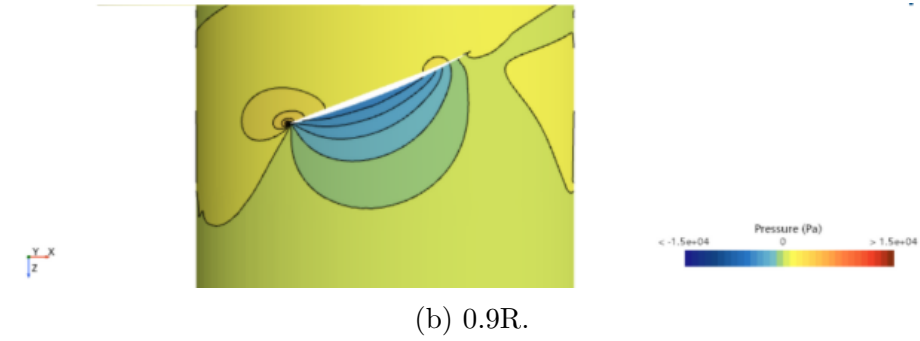
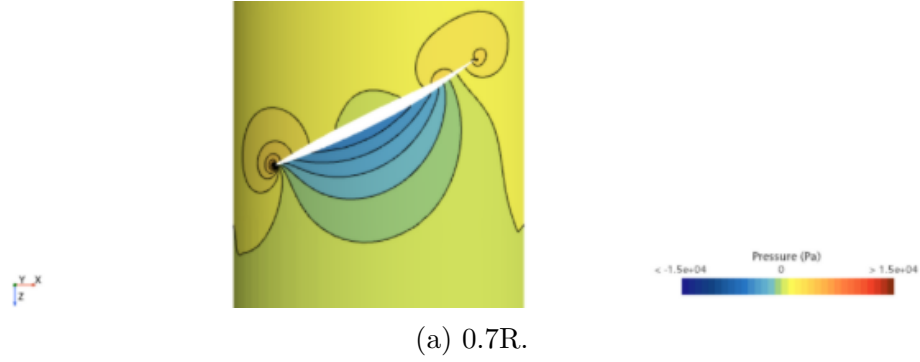


Figure 37: Pressures at two radii of the 4119.

From Figure 36, the possibility of cavitation can be seen from the high velocity at the tip of the blade. By taking radial section cuts like Figure 37, it can be ensured that each airfoil is acting properly by produce a high pressure area behind the blade (Figure 21) indicating that they are creating thrust in the outflow direction.

4 Optimization of Toroidal Propellers

4.1 Initial Propellers

The profiles of the 4119 screw propeller and the TORUS 0000 toroidal propeller are shown below.

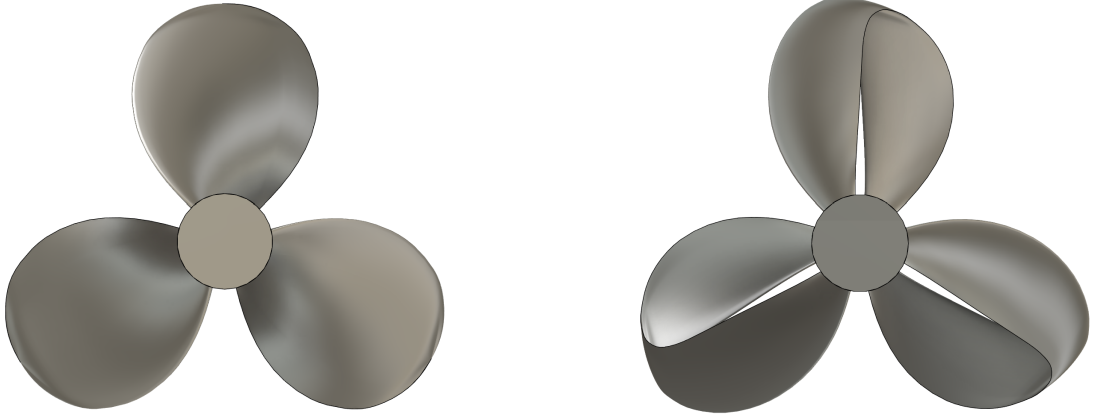


Figure 38: Front views of the 4119 and the TORUS 0000 toroidal propeller.

Both propellers were run through the designed simulation and the results for K_T , $10K_Q$, and η_0 were determined and are outlined in Table 12. At the original run conditions of the 4119, the toroidal version was able to generate more thrust; however, that came with the cost of a higher required torque. This means that the initial toroidal propeller is less efficient than the base screw propeller.

Table 12: CFD Performance Comparison of 4119 and TORUS 0000

Propeller	K_T	$10K_Q$	η_0
4119	0.155	0.294	0.701
0000	0.167	0.355	0.623

A comparison of the pressure distributions across the screw and toroidal propeller blades as seen in Figure 39 indicated the cause of the toroidal propeller's lower efficiency. The 4119 displays a well-defined pressure differential, with a low-pressure region upstream (suction side) and a high-pressure region downstream (pressure side). This pressure gradient is what generates the thrust produced by the rotation of the propeller.

The pressure differential across the TORUS 0000 toroidal propeller blade is significantly weaker compared to that of the 4119 screw propeller. The suction side of the toroidal propeller has much higher pressure, and the pressure downstream is not significantly changed compared to upstream. This suggests increased resistance to

rotation, manifesting as a higher required torque without a proportional increase in thrust. This significantly reduces the toroidal propeller's efficiency.

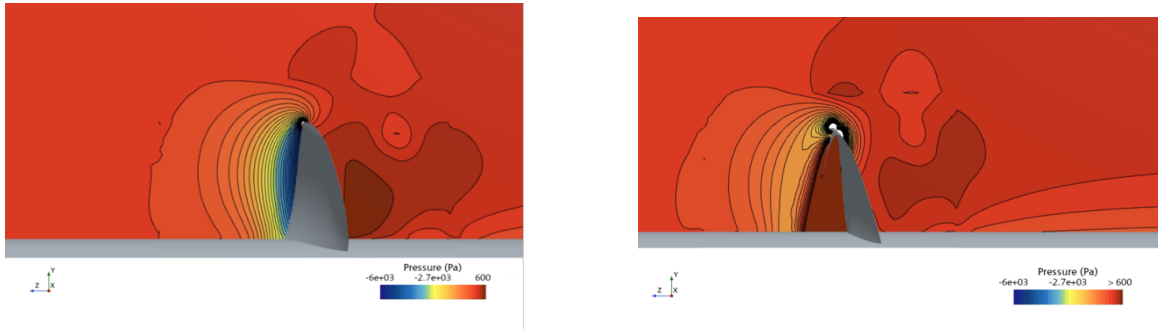


Figure 39: Pressure across the blades of the 4119 and the TORUS 0000 toroidal propeller.

To better understand the cause of the weaker pressure differential, radial cross sections of the propeller blade were taken in Figure 42, isolating the airfoil sections that form the blade. The upstream section of the blade shows a low-pressure region above the airfoil and a high-pressure region beneath, which is desirable for thrust generation.

However, the rear blade section of the blade does not have a well-defined pressure differential across the airfoil. Instead, there are pockets of high-pressure above the airfoil and low-pressure underneath. This observation is a driving factor in following propeller iterations, as propeller parameters were altered in order to develop properly functioning airfoils with optimal pressure differentials.

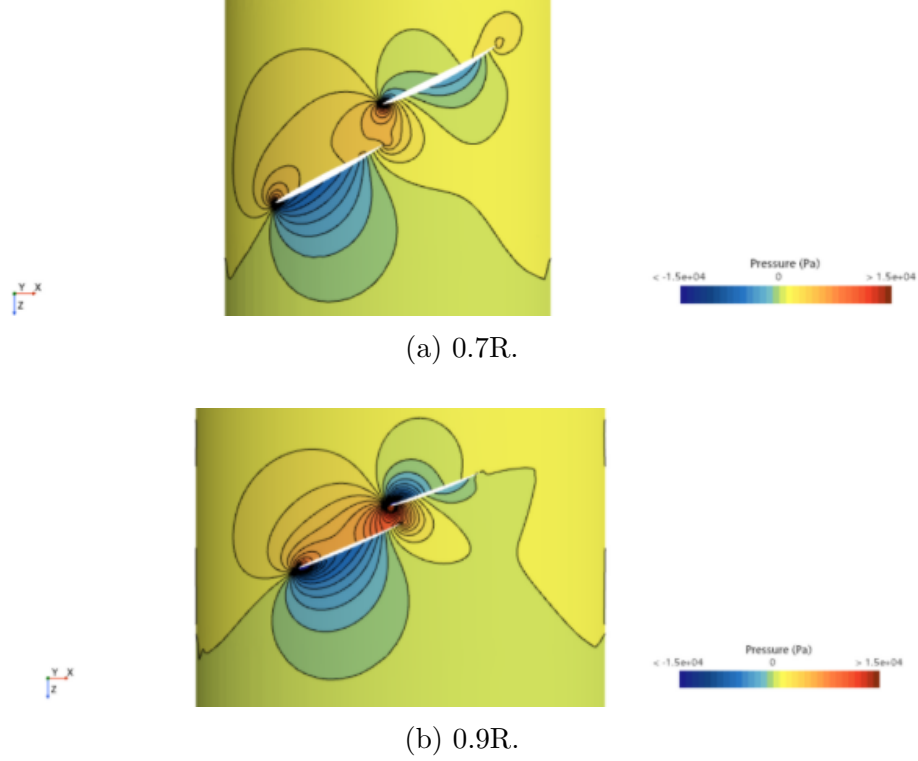


Figure 40: Pressures at 0.7 and 0.9R for the TORUS 0000.

4.2 Manual Optimizations

It can be observed from the differences of the screw and toroidal propeller reference lines that l and ϕ will play a large role in the shape of the propeller. As they are derived from the 4119 propeller itself, this work keeps airfoil parameters, f and t , and blade parameters θ_s and k constant. Since this comparison is also based around a consistent EAR, the c found in Section 2.4.3 is held constant. Holding these parameters constant allows direct correspondence from a screw propeller to a generated toroidal propeller. Reducing the number of changing parameters decreases the available search space although allows for easier development of a similar propeller.

The blade parameters l , ϕ , and α are all originally from [15] and ψ is the modified values from Section 2.4.3. These four parameters along with the rear blade section of P are the suggested parameters to modify. . The naming scheme used in this work is a four number code that describes the distribution of the changed parameters. Each number corresponds to the following parameter: l , ϕ , P , and α . During a comparison, these parameters should be changed as they are the newest additions, while during optimization all parameters should be altered to increase performance in every aspect of the search space.

4.2.1 Wheelbase Distributions

Two variations of the l/L distribution were created, both of which are independent from other sources. l/L distribution 1 is an inverse quadratic function of radius seen in Equation 44, while l/L distribution 2 is an inverse quartic function seen in Equation 45. Quadratic and quartic functions were chosen to provide a simple starting point. These distributions reach $l/L = 0.5$ at $r/R = 1$ as opposed to l/L distribution 0 which reaches $l/L = 0.6$. Attempts to develop a piecewise quadratic function that reached $l/L = 0.6$ at $r/R = 1$ were unsuccessful, although may be able to be created by altering other parameters.

$$l/L\left(\frac{r}{R}\right) = \begin{cases} 0.5 - \left(-\frac{5}{16} \left(\frac{r}{R} - 1\right)\right)^{0.5}, & \frac{r}{R} = [0.2 \rightarrow 1] \\ 0.5 + \left(-\frac{5}{16} \left(\frac{r}{R} - 1\right)\right)^{0.5}, & \frac{r}{R} = [1 \rightarrow 0.2] \end{cases} \quad (44)$$

$$l/L\left(\frac{r}{R}\right) = \begin{cases} 0.5 - \left(-\frac{5}{64} \left(\frac{r}{R} - 1\right)\right)^{0.25}, & \frac{r}{R} = [0.2 \rightarrow 1] \\ 0.5 + \left(-\frac{5}{64} \left(\frac{r}{R} - 1\right)\right)^{0.25}, & \frac{r}{R} = [1 \rightarrow 0.2] \end{cases} \quad (45)$$

Table 13: Various Wheelbase Distributions

r/R	l/L Dist. 0	l/L Dist. 1	l/L Dist. 2
0.200	0.0000	0.0000	0.0000
0.300	0.0624	0.0323	0.0164
0.400	0.1203	0.0670	0.0347
0.500	0.1773	0.1047	0.0554
0.600	0.2343	0.1464	0.0796
0.700	0.2937	0.1938	0.1087
0.800	0.3590	0.2500	0.1464
0.900	0.4377	0.3232	0.2027
0.950	0.4889	0.3750	0.2500
0.975	0.5250	0.4116	0.2898
0.995	0.5729	0.4605	0.3594
1.000	0.6000	0.5000	0.5000
0.995	0.6438	0.5395	0.6406
0.975	0.6842	0.5884	0.7102
0.950	0.7146	0.6250	0.7500
0.900	0.7574	0.6768	0.7973
0.800	0.8150	0.7500	0.8536
0.700	0.8582	0.8062	0.8913
0.600	0.8941	0.8536	0.9204
0.500	0.9249	0.8953	0.9446
0.400	0.9523	0.9330	0.9653
0.300	0.9771	0.9677	0.9836
0.200	1.0000	1.0000	1.0000

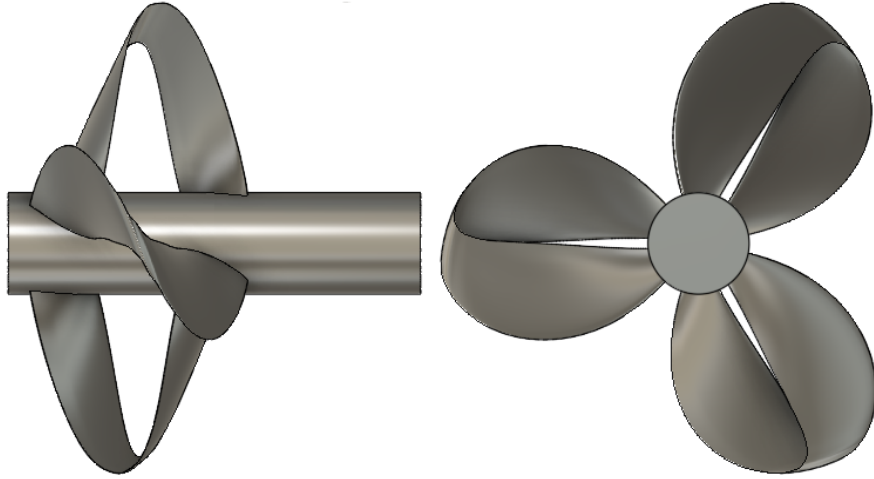


Figure 41: TORUS 1000 propeller as per Table 13.

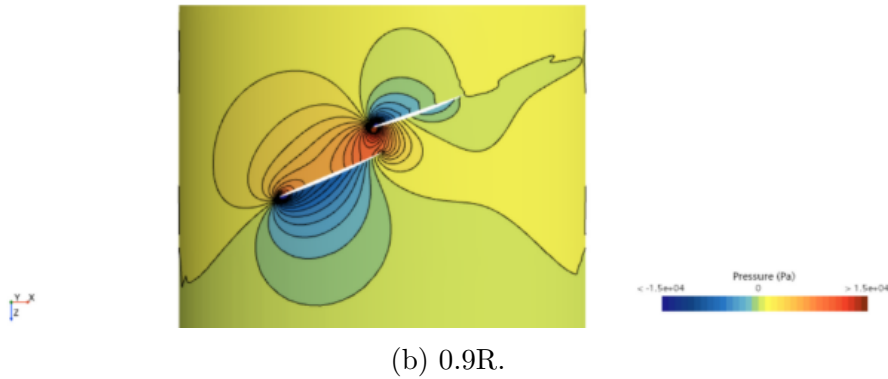
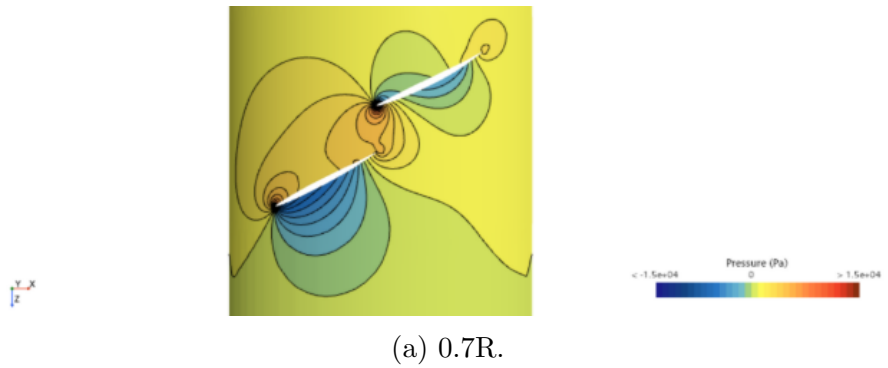


Figure 42: Pressures at 0.7R and 0.9R for the TORUS 1000.

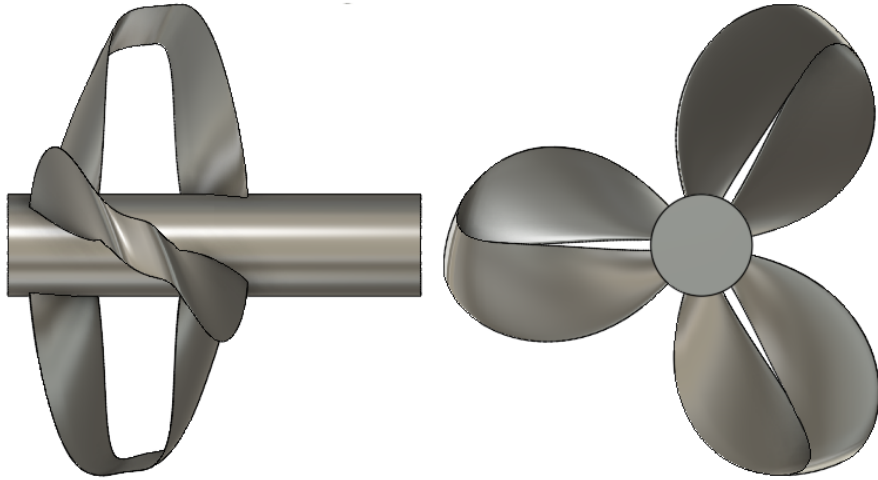
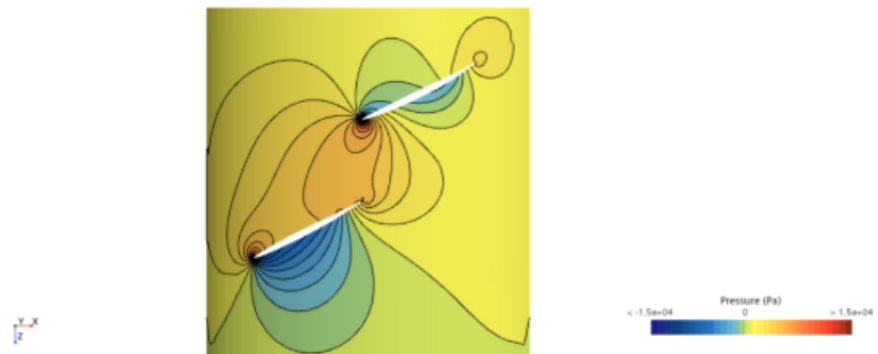
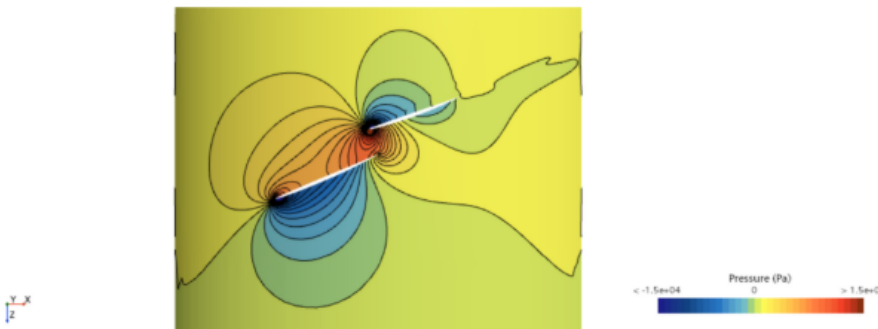


Figure 43: TORUS 2000 propeller as per Table 13.



(a) 0.7R.



(b) 0.9R.

Figure 44: Pressures at 0.7R and 0.9R for the TORUS 2000.

As seen in Figures 42 and 44, there is still high pressure on the wrong side of the rear blade airfoil, which continues to cause the higher torque. $10K_Q$ was 0.359 for TORUS 1000 and 0.458 for TORUS 2000, demonstrating that $10K_Q$ is proportional

to wheelbase. Like torque, thrust was also proportional to wheelbase changes, as TORUS 1000 had a $K_T = 0.165$ and TORUS 2000 had 0.184. Unfortunately, the loss due to the torque is more significant than the thrust gains, leading to an efficiency of 0.608 for TORUS 1000 and 0.491 for TORUS 2000, both of which are lower than TORUS 1000.

4.2.2 Skew Distributions

Two variations of the skew (ϕ) distribution were created, both scales of ϕ distribution 0. ϕ distribution 1 increased ϕ distribution 0 by 20% while ϕ distribution 2 halved it. While no independent distributions were created, skew is very symmetric and a function could reasonably be fit to approximate it. These scalings were an attempt to separate the front and rear sections of the blade to improve flow.

Table 14: Various Reference Line Skew Distributions

r/R	ϕ° Dist. 0	ϕ° Dist. 1	ϕ° Dist. 2
0.200	-27.56	-33.07	-13.78
0.300	-25.52	-30.62	-12.76
0.400	-23.44	-28.13	-11.72
0.500	-21.26	-25.51	-10.63
0.600	-18.93	-22.72	-9.47
0.700	-16.33	-19.60	-8.17
0.800	-13.26	-15.91	-6.63
0.900	-9.33	-11.20	-4.67
0.950	-6.55	-7.86	-3.28
0.975	-4.56	-5.47	-2.28
0.995	-1.86	-2.23	-0.93
1.000	-0.24	-0.29	-0.12
0.995	2.45	2.94	1.23
0.975	4.99	5.99	2.50
0.950	6.95	8.34	3.48
0.900	9.78	11.74	4.89
0.800	13.77	16.52	6.89
0.700	16.83	20.20	8.42
0.600	19.44	23.33	9.72
0.500	21.74	26.09	10.87
0.400	23.82	28.58	11.91
0.300	25.74	30.89	12.87
0.200	27.56	33.07	13.78

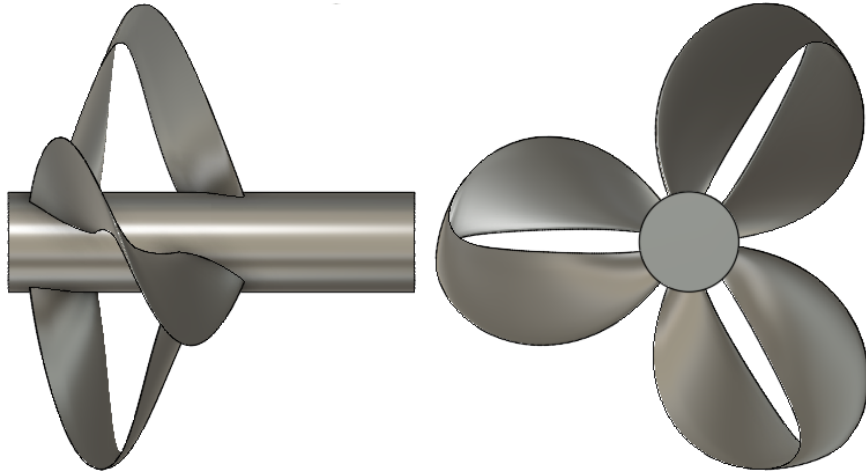
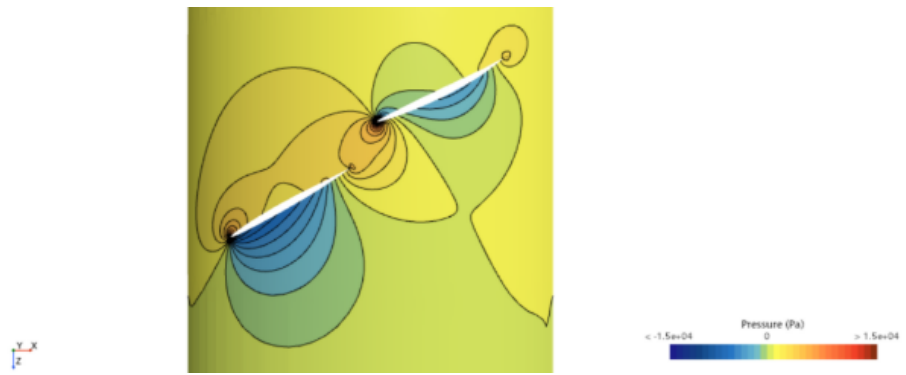
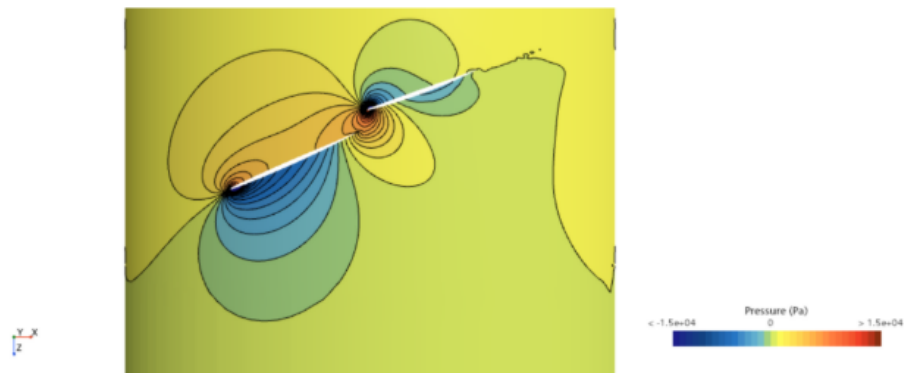


Figure 45: TORUS 0100 propeller as per Table 14.



(a) 0.7R



(b) 0.9R

Figure 46: Pressures at 0.7R and 0.9R for the TORUS 0100

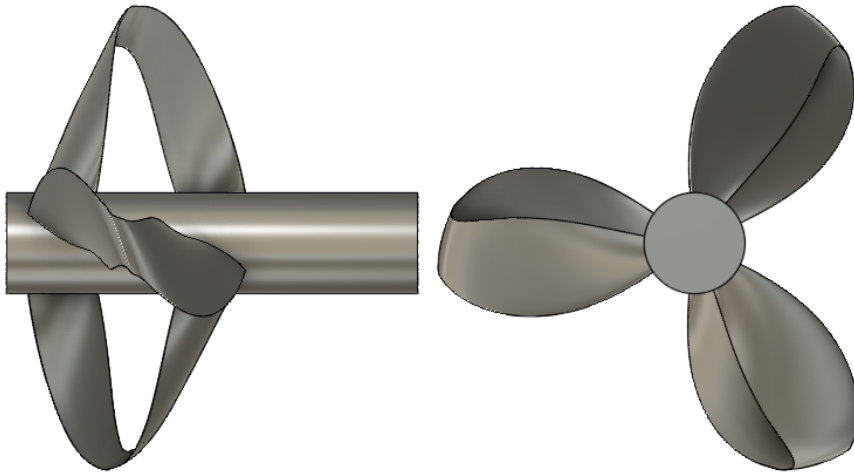
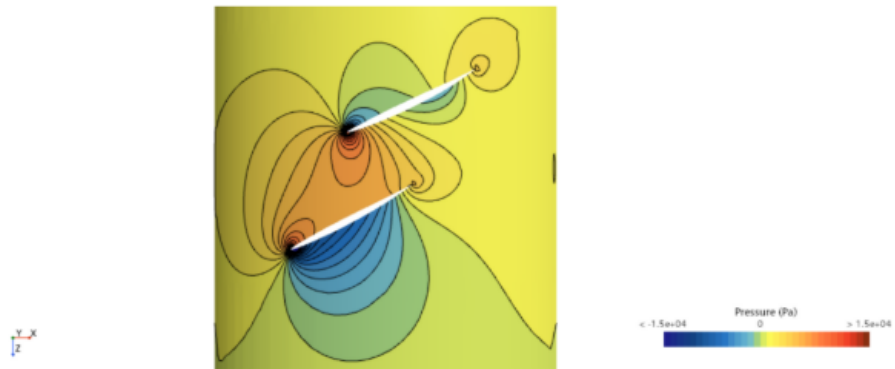
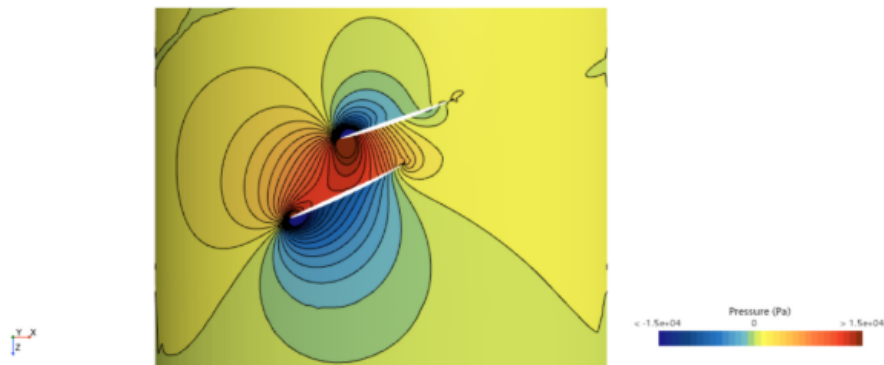


Figure 47: TORUS 0200 propeller as per Table 14.



(a) 0.7R.



(b) 0.9R.

Figure 48: Pressures at 0.7R and 0.9R for the TORUS 0200.

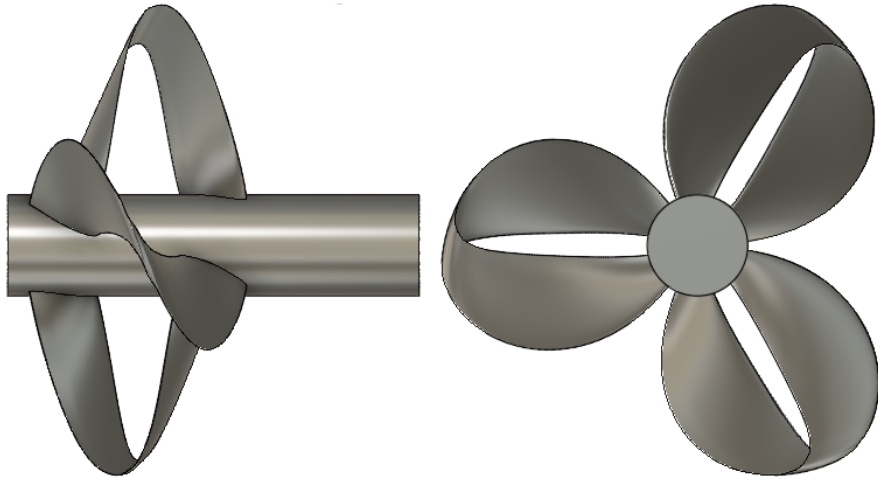


Figure 49: TORUS 1100 propeller as per Table 14.

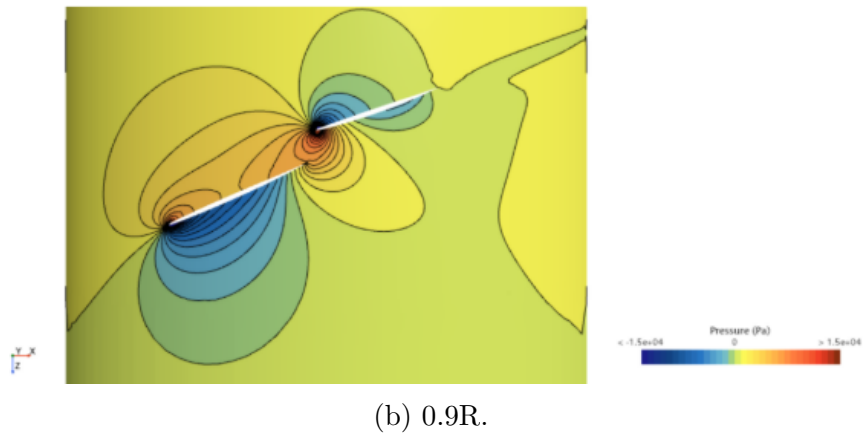
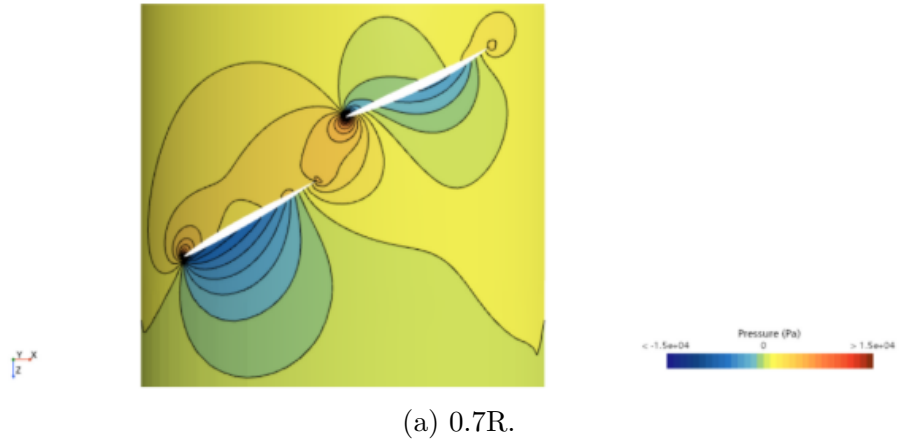


Figure 50: Pressures at 0.7R and 0.9R for the TORUS 1100.

The first modification to the EAR-matched baseline, TORUS 0000, involved increasing the ϕ distribution by 20%, resulting in TORUS 0100. This change led to a small decrease in thrust coefficient, K_T , from 0.167 to 0.166, but also slightly reduced torque, $10K_Q$, from 0.355 to 0.339, yielding an efficiency of 0.650, the highest efficiency observed among the tested configurations. When the ϕ distribution was halved to create TORUS 0200, K_T and $10K_Q$ increased to 0.180 and 0.538, respectively, resulting in an efficiency of 0.538. TORUS 1100 saw a large increase in efficiency when compared to TORUS 1000 ($\eta_0 = 0.637$). It had a similar thrust coefficient, but saw a large decrease in torque.

This indicates that changes in ϕ distribution affect thrust and torque differently, with torque being more sensitive to variations. Increasing skew slightly reduces thrust, but leads to a much larger reduction in torque, resulting in improved efficiency. Higher ϕ distributions provide a better balance between thrust generation and required torque.

4.2.3 Pitch Distributions

To maintain an accurate conversion of a screw propeller to a toroidal propeller, pitch must be altered with care. In the baseline TORUS 0000, the downstream blade section exhibited a weakened pressure differential, indicating insufficient local angle of attack and reduced loading. Altering the pitch distribution in the iterations described below aimed to correct this phenomena, and create realistic pressure differentials across the cross sectional airfoils that make up the toroidal propeller blade.

The three distributions created are all scaled from pitch distribution 0. Pitch distribution 1 was an attempt at scaling the rear blade section by a smoothed function fit to the distribution found in Ye et al. Pitch distribution 2 shows the effect of scaling the entirety of the distribution, it is also scaled by the Ye et al. distribution centered around $r/R = 0.7$. Pitch distribution 3 accounts for the altered inflow by scaling the local angle of attack on the rear section of the blade. Apparent inflow velocity is calculated from the speed at which the propeller is rotation (Equation 46). The inflow direction is then found and subtracted from the geometric angle of attack (Equation 47). This angle of attack is increased by a multiplier and the reverse process can easily find P/D . Attempts to create TORUS 1030 were unsuccessful, although a change in the α distribution resulted in a successful TORUS 1031. Due to the rotations of the airfoils around the tip of the propeller, α may act similarly to traditional pitch at the tip.

$$V_{tangent} = \pi \frac{r}{R} D n \quad (46)$$

$$\beta_{actual} = \beta - \arctan(V_{inlet}/V_{tangent}) \quad (47)$$

Table 15: Various Pitch Distributions and Geometric Angle of Attack Multiplier

r/R	P/D Dist. 0	P/D Dist. 1	P/D Dist. 2	P/D Dist. 3	Multiplier
0.200	1.105	1.105	1.187	1.105	1.0000
0.300	1.102	1.102	1.204	1.102	1.0000
0.400	1.098	1.098	1.209	1.098	1.0000
0.500	1.093	1.093	1.195	1.093	1.0000
0.600	1.088	1.088	1.159	1.088	1.0000
0.700	1.084	1.084	1.084	1.084	1.0000
0.800	1.081	1.081	0.976	1.083	1.0092
0.900	1.079	1.079	0.843	1.096	1.0677
0.950	1.077	1.077	0.781	1.123	1.1839
0.975	1.076	1.076	0.751	1.152	1.3033
0.995	1.076	1.076	0.730	1.190	1.4524
1.000	1.075	1.075	0.729	1.200	1.5000
0.995	1.076	1.135	0.772	1.214	1.5476
0.975	1.076	1.239	0.861	1.252	1.6967
0.950	1.077	1.303	0.941	1.286	1.8161
0.900	1.079	1.359	1.070	1.322	1.9323
0.800	1.081	1.391	1.263	1.345	1.9908
0.700	1.084	1.396	1.396	1.360	2.0000
0.600	1.088	1.394	1.481	1.377	2.0000
0.500	1.093	1.388	1.510	1.401	2.0000
0.400	1.098	1.381	1.505	1.433	2.0000
0.300	1.102	1.370	1.495	1.476	2.0000
0.200	1.105	1.358	1.476	1.538	2.0000

Table 16: Various Vertical Angle Distributions

r/R	α° Dist. 0	α° Dist. 1
0.200	0.00	0.00
0.300	0.12	1.12
0.400	0.62	1.62
0.500	1.44	2.44
0.600	2.61	3.61
0.700	4.37	5.37
0.800	6.89	7.89
0.900	9.39	10.39
0.950	10.45	11.45
0.975	10.98	11.98
0.995	11.41	12.41
1.000	11.51	12.51
0.995	11.28	12.28
0.975	10.52	11.52
0.950	8.97	9.97
0.900	5.54	6.54
0.800	2.33	3.33
0.700	1.44	2.44
0.600	0.98	1.98
0.500	0.63	1.63
0.400	0.38	1.38
0.300	0.17	1.17
0.200	0.00	0.00

TORUS 2010 was generated using l/L distribution 1 (Table 13) and pitch distribution 1 (Table 15). The modified pitch produces a clearer pressure gradient, with high pressure on the pressure side and low pressure on the suction side, improving upon earlier iterations (TORUS 0000) where downstream sections showed weak pressure differentials. However, the pitch change is small, and the wheelbase distribution increases diagonal spacing between the front and rear blade sections. A low-pressure region persists upstream of the leading airfoil, indicating that the downstream section still operates in the upstream blade wake. This suggests that the local inflow angle and effective angle of attack remain suboptimal across the toroidal loop.

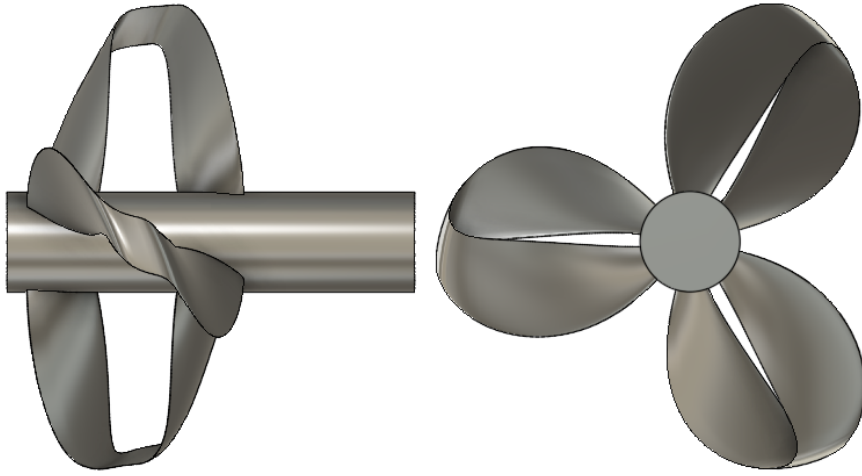


Figure 51: TORUS 2010 propeller as per Table 15.

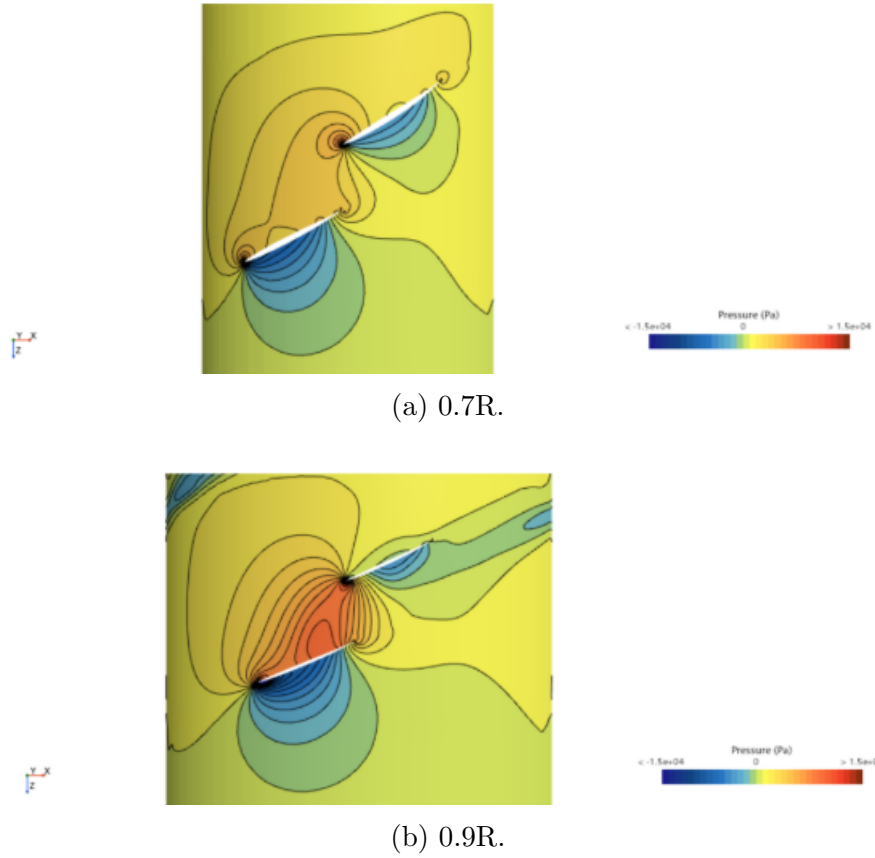


Figure 52: Pressures at 0.7R and 0.9R for the TORUS 2010.

TORUS 0020 was created with pitch distribution 2 following Table 15. The propeller profile and its corresponding pressure distribution are shown in Figures 51

and 54, respectively. As with the previous variants shown (0000 and 2010), the upstream airfoil presents a clear pressure differential with a low-pressure region along the upper surface and a corresponding high-pressure region beneath the lower surface, consistent with the profile required to generate thrust. In the downstream airfoil, a low-pressure region is present along the upper surface, showing an improvement from TORUS 0000, which presented pockets of high-pressure above this airfoil. Additionally, TORUS 0020 presents a drastic pressure differential on both airfoils compared to TORUS 2010 presented above, which showed a more diffuse distribution that can be attributed to the flow interactions occurring between the two airfoils. These differences contribute to the greater efficiency obtained with 0020 ($\eta_0 = 0.626$) compared to 0000 ($\eta_0 = 0.623$) and 2010 ($\eta_0 = 0.560$).

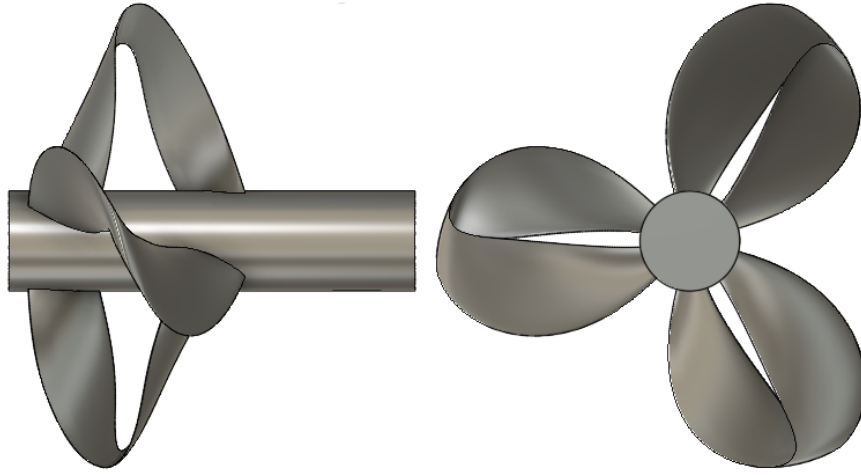
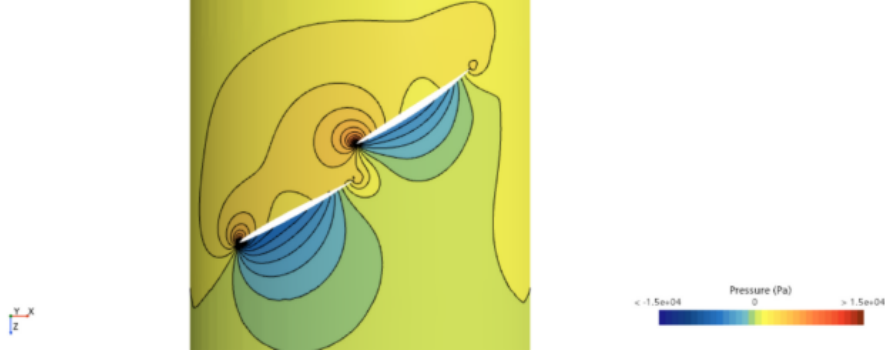
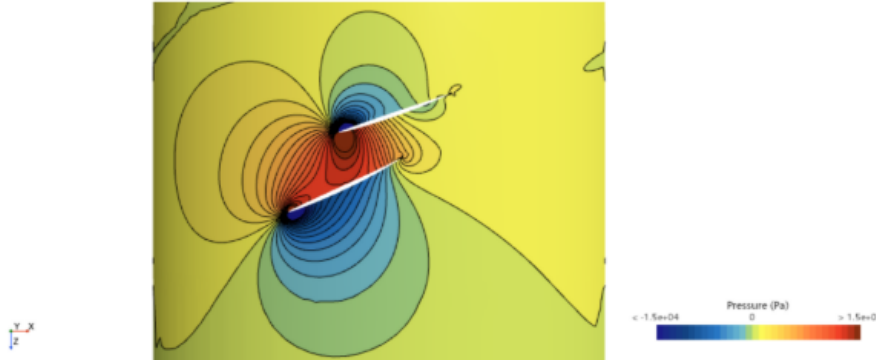


Figure 53: TORUS 0020 propeller as per Table 15.



(a) 0.7R.



(b) 0.9R.

Figure 54: Pressures at 0.7R and 0.9R for the TORUS 0020.

TORUS 1020 was created using l/L distribution 1 in Table 13 and pitch distribution 2 in Table 15. The propeller model and the pressure distribution obtained are shown in Figures 55 and 56, respectively. Modifying the wheelbase with distribution 1 produced nearly identical pressure distributions across the airfoils. Small differences can be seen in the profile of the downstream airfoil, particularly at the trailing edge, where a small pocket of low pressure is formed at the lower surface, which is not present in 0020. In performance analysis, it is seen that 1020 obtains an efficiency of $\eta_0 = 0.620$, lower than the efficiency of $\eta_0 = 0.626$ obtained with 0020.

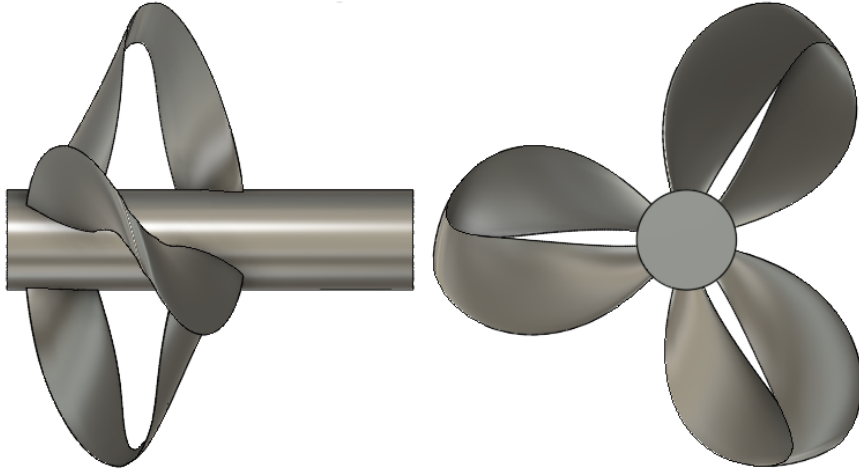
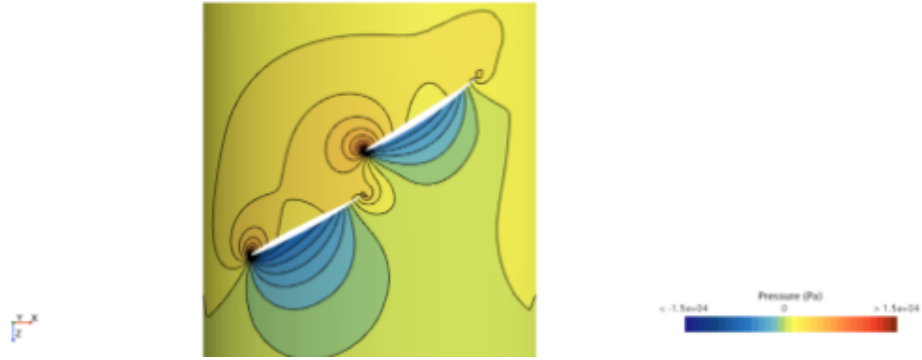
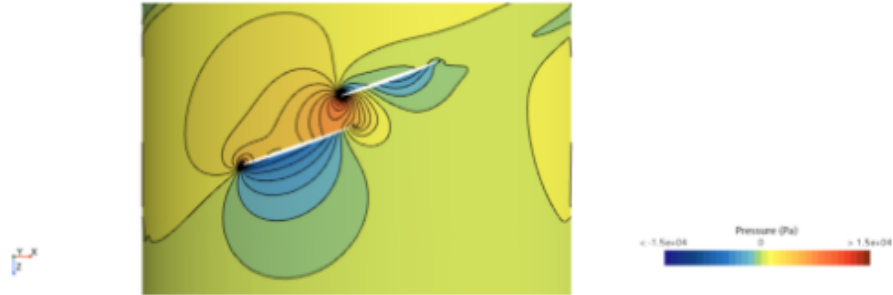


Figure 55: TORUS 1020 propeller as per Table 15.



(a) 0.7R.



(b) 0.9R.

Figure 56: Pressures at 0.7R and 0.9R for the TORUS 1020.

TORUS 1120 was created using l/L distribution 1 in Table 13, ϕ distribution 1 in Table 14, and pitch distribution 2 in Table 15. The propeller model and the pressure distribution obtained are shown in Figures 57 and 58, respectively. The pressure

distribution across the airfoils displays a low-pressure region upstream and a high-pressure region downstream on both front and rear airfoils at a 0.7R. TORUS 1120 achieves an efficiency of $\eta_0 = 0.630$, lower than the efficiency of $\eta_0 = 0.637$ produced with 1100, despite the corrected airfoil pressure distribution.

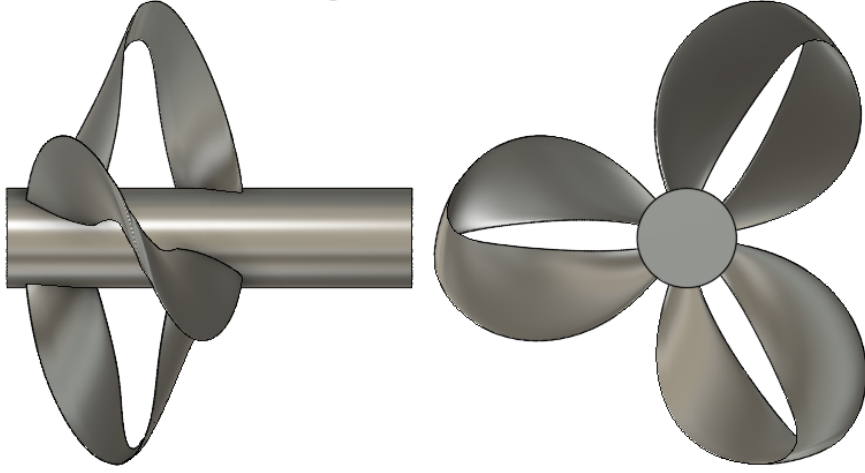
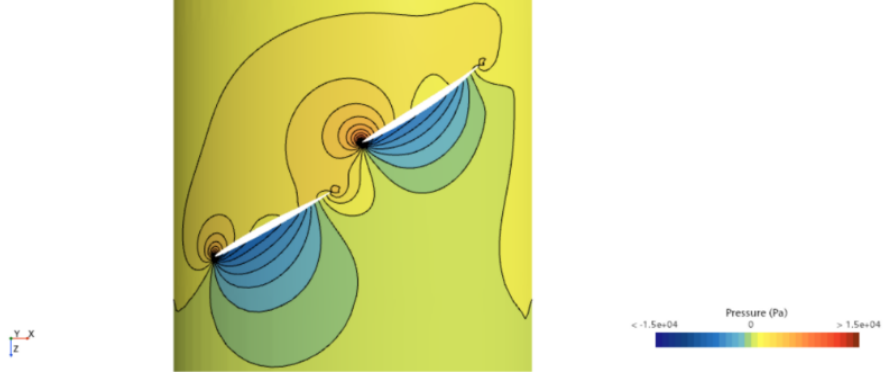
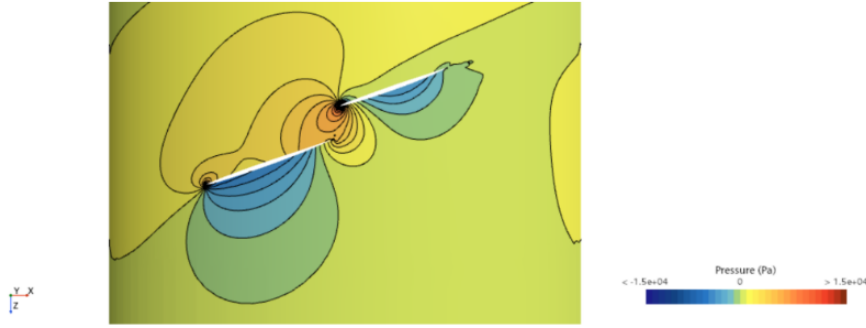


Figure 57: TORUS 1120 propeller as per Table 15.



(a) 0.7R.



(b) 0.9R.

Figure 58: Pressures at 0.7R and 0.9R for the TORUS 1120.

TORUS 2020 was created using l/L distribution 2 in Table 13 and pitch distribution 2 in Table 15. The propeller model and the pressure distribution obtained are shown in Figures 59 and 60, respectively. Modifying the wheelbase with distribution 2 produced a more concentrated low-pressure region in the downstream airfoil, with more symmetric contours, suggesting improved flow attachment and reduced flow interference between the upstream and downstream airfoils. This improved pressure distribution is reflected in the performance of the propeller, as it is able to generate increased thrust ($K_T = 0.222$) when compared to 0020 ($K_T = 0.220$) and 1020 ($K_T = 0.219$). However, TORUS 2020 presents considerably lower efficiency ($\eta_0 = 0.537$) due to its greater torque ($10K_Q = 0.549$) as a result of the aggressive change in wheelbase.

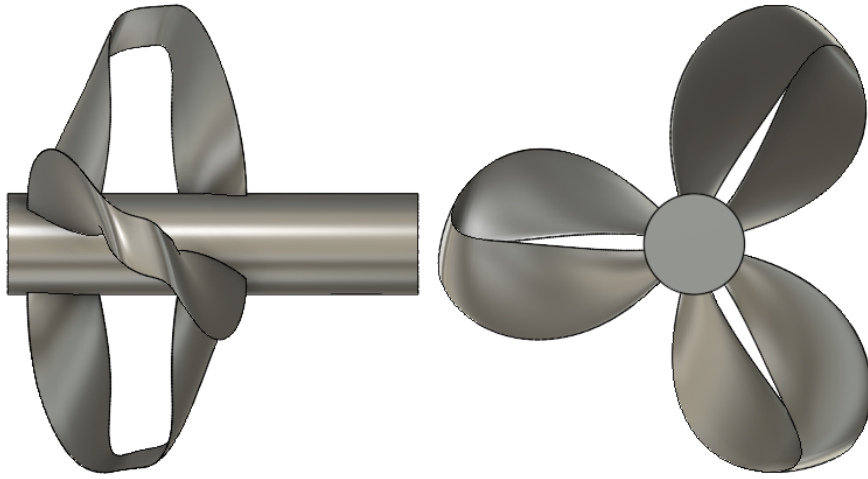


Figure 59: TORUS 2020 propeller as per Table 15.

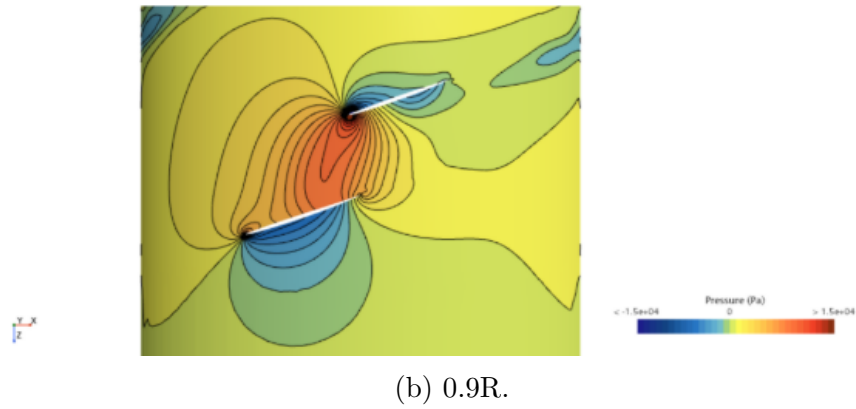
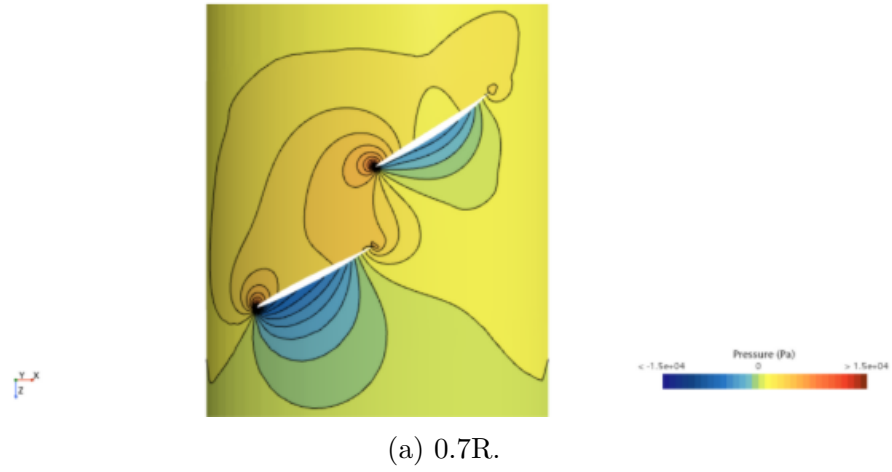


Figure 60: Pressures at 0.7R and 0.9R for the TORUS 2020.

TORUS 1031 was created using l/L distribution 1 in Table 13, pitch distribution

3 in Table 15, and vertical angle distribution 1 in Table 16. The propeller model and the pressure distribution obtained are shown in Figures 61 and 62, respectively. As with previous variants, the TORUS 1031 generates a well-defined pressure differential in the upstream airfoil. In the downstream airfoil, asymmetries arise in the low-pressure region resulting from wake interactions with the upstream airfoil. However, this downstream airfoil is still able to generate a pressure profile that allows for flow attachment and thrust generation.

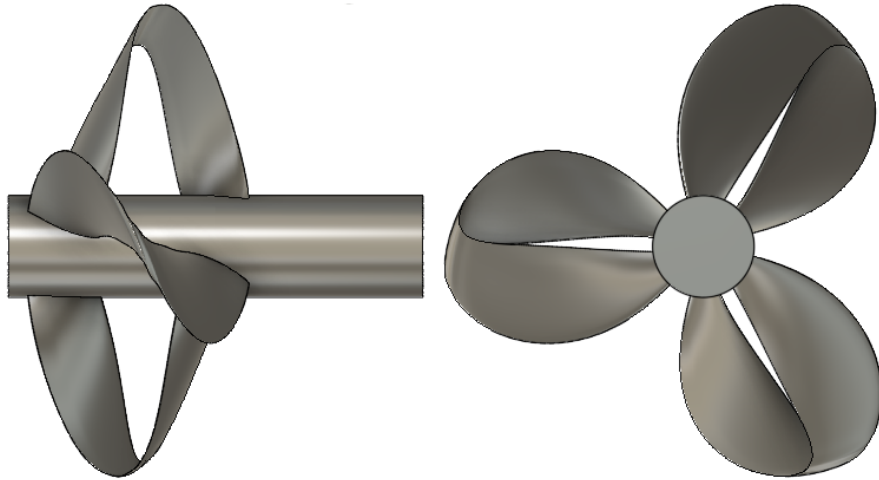
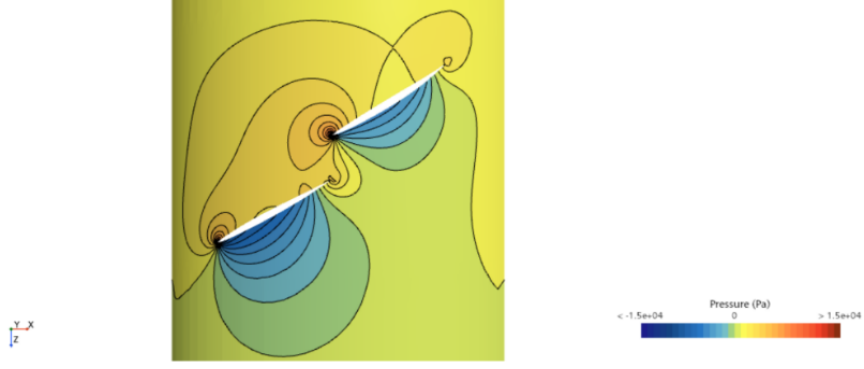
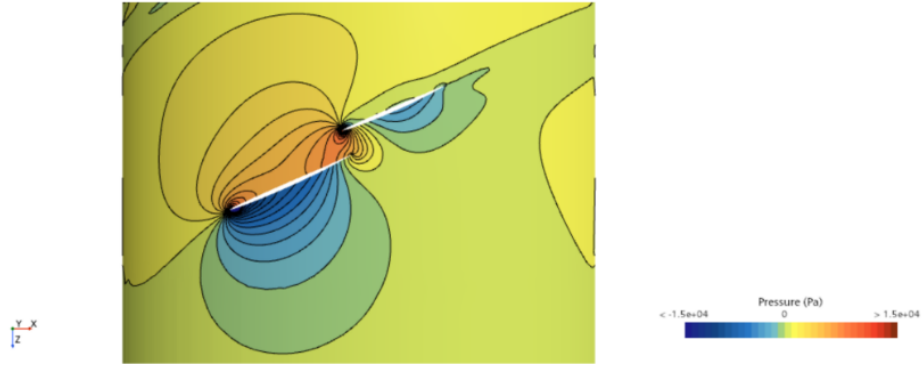


Figure 61: TORUS 1031 propeller as per Table 15 and Table 16.



(a) 0.7R.



(b) 0.9R.

Figure 62: Pressures at 0.7R and 0.9R for the TORUS 1031.

Pitch modifications restored the pressure differential and significantly increased thrust coefficient while maintaining efficiency, demonstrating that increasing pitch in the rear blade section is necessary to recover aerodynamic loading in toroidal geometries. Pitch adjustments, aimed at controlling local inflow conditions, show a clear benefit. For example, TORUS 1020 achieved $\eta_0 = 0.620$ with torque $10K_Q = 0.469$, partially recovering efficiency lost from the wheelbase change while moderating torque. Similar behavior is seen in TORUS 0020, where pitch modification raised efficiency to $\eta_0 = 0.626$ with $10K_Q = 0.465$. These results demonstrate that regulating the local inflow along the blade is necessary for maintaining performance after geometric changes.

Combined modifications further show this. TORUS 1031 reached $\eta_0 = 0.634$, while TORUS 1120 achieved $\eta_0 = 0.630$ with lower torque ($10K_Q = 0.460$). This indicates that coordinating skew, wheelbase, and pitch can recover efficiency lost to individual parameter changes while controlling torque growth, highlighting the importance of multi-parameter optimization.

5 Discussion

Of the propellers that were created, none were able to match the 4119's efficiency of 0.701 (Figure 63). TORUS 0100 had the largest efficiency of 0.650 and the smallest torque coefficient of 0.339. TORUS 2010 had the largest thrust coefficient of 0.270 but also had the largest torque coefficient of 0.638. However, all toroidal propellers produced more thrust than the 4119, likely due to the additional blade section.

Changing the wheelbase alone saw declines in performance when compared to l/L distribution 0. ϕ distribution 1 saw increases in performance by decreasing the required thrust. Increasing the rear blade section pitch saw universal increases in efficiency when compared to pitch distribution 0.

Table 17: Performance of Designed Propellers

Propeller	k_T	$10k_Q$	η_0
4119	0.155	0.294	0.701
0000	0.167	0.355	0.623
0100	0.166	0.339	0.650
1000	0.165	0.359	0.608
2000	0.184	0.498	0.491
2010	0.270	0.638	0.560
1020	0.219	0.469	0.620
2020	0.222	0.549	0.537
0200	0.180	0.443	0.538
1031	0.252	0.526	0.634
0020	0.220	0.465	0.626
1100	0.161	0.334	0.637
1120	0.218	0.460	0.630

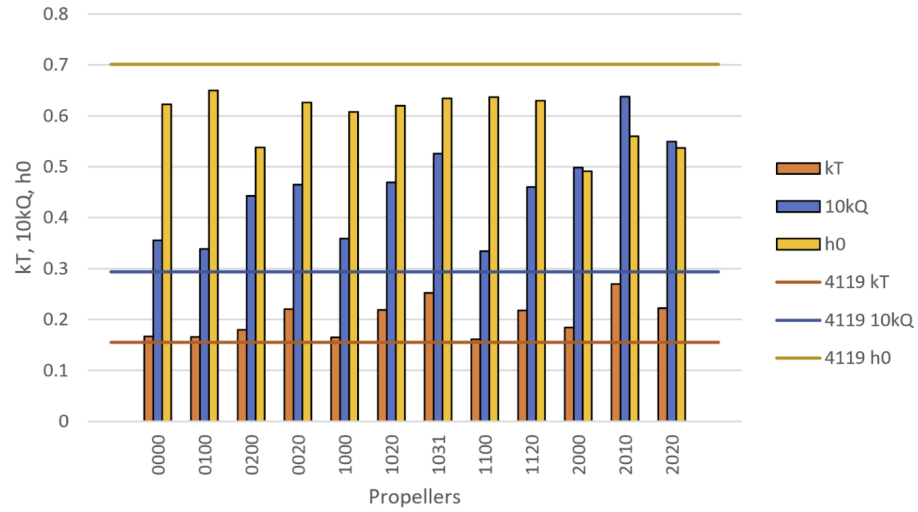


Figure 63: Comparison of all TORUS propellers and the 4119

One of the main issues with the TORUS propellers comes from the interaction between the front and rear blades. The wake of the front blade often hits the rear blade causing a force acting opposite to the desired force (Figure 64). However, if the blades were separated to avoid this, the tip region became quite large leading to large low pressure zones at the tip indicating cavitation (Figure 65).

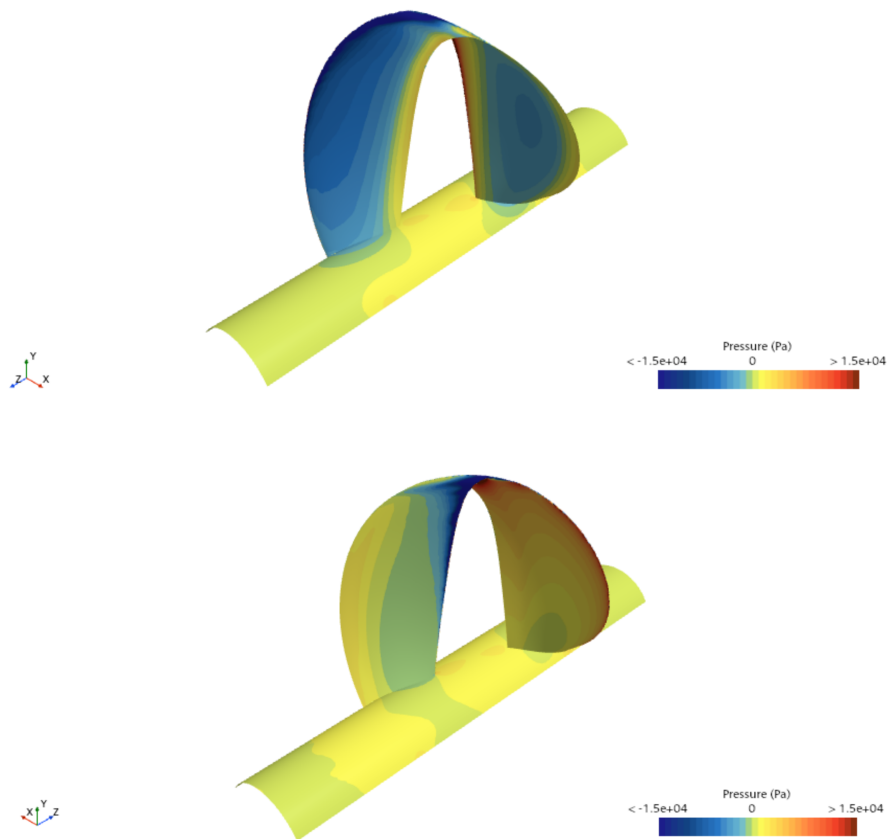


Figure 64: Pressure on both sides of TORUS 0100.

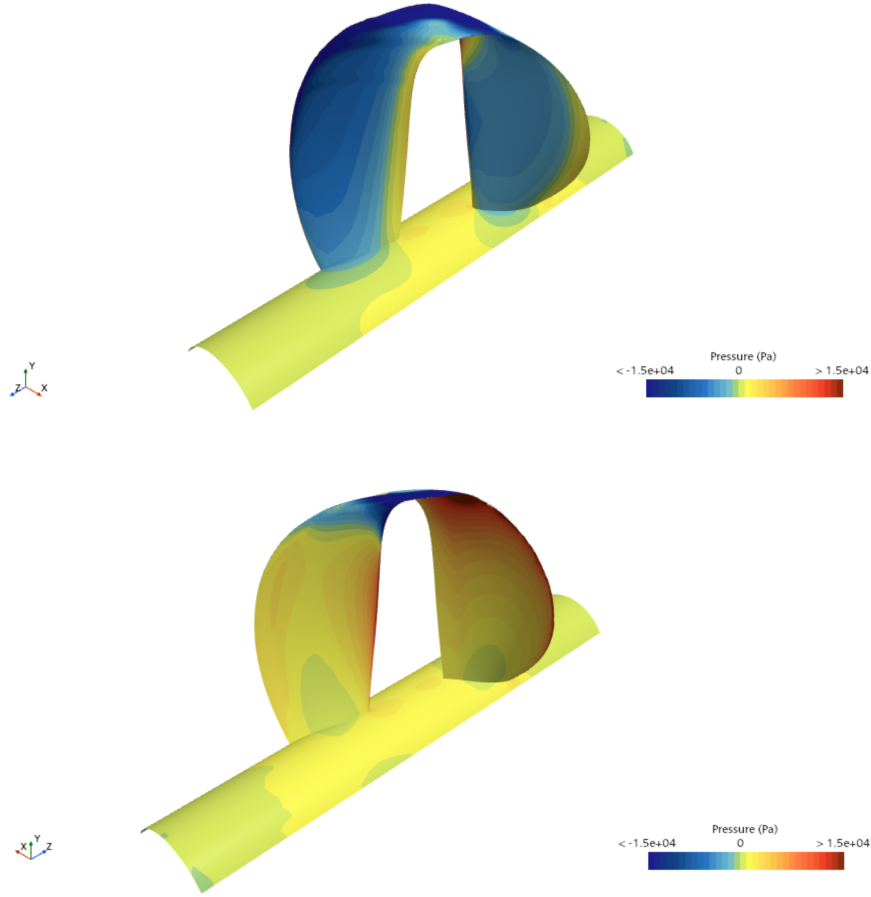
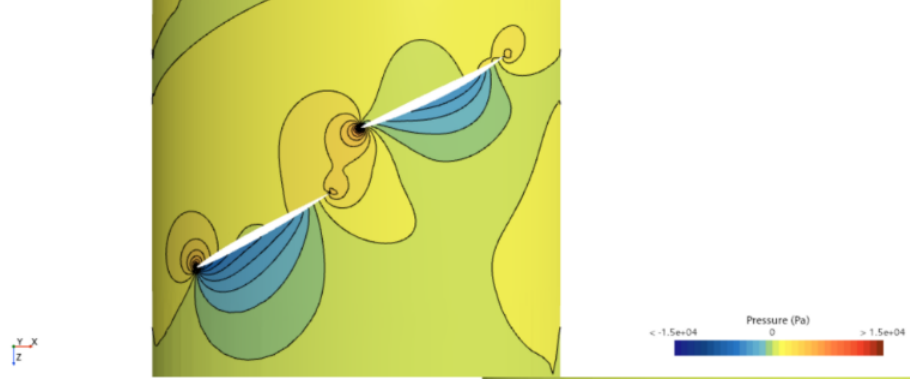
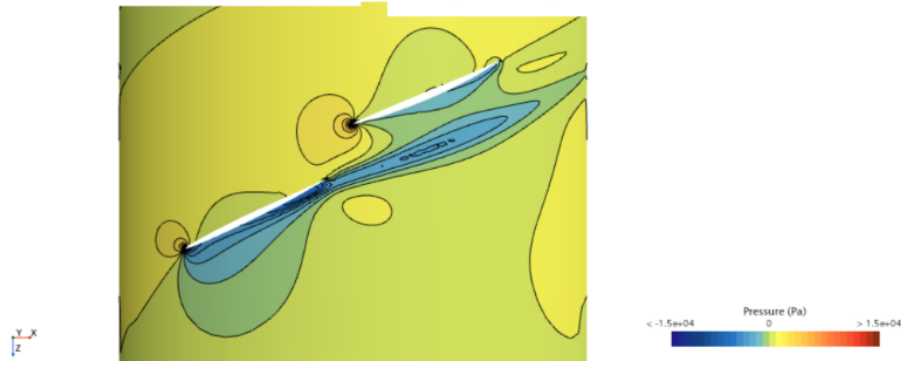


Figure 65: Pressure on both sides of TORUS 2010.

Since the tip region of the blade is where most of the negative interactions occur, the tip of TORUS 1100 was removed and simulated. The resulting propeller had a significant reduction in torque from the original ($10k_Q = 0.193$), which shows that most of the drag was in fact caused by the tip of the propeller. The thrust, like the torque, also had a reduction to 0.0993. This is likely because this propeller has the proper pressure gradient along each airfoil (Figure 66). This led to an efficiency of 0.681, making it the closest to the 4119 in performance. This propeller also has major physical drawbacks since it has two narrow blades, which can likely induce twice the tip vortices.



(a) 0.7R.



(b) 0.79R.

Figure 66: Pressures at 0.7R and 0.79R for the tipless TORUS 1100.

Another issue is that creating a standard reference line proves to be difficult as the connectivity of the blade is highly dependent on the combination of many rotational parameters. As seen earlier, the ψ distribution needed to be changed for the propeller to successfully generate. While this set of parameters that were used successfully generated multiple propellers, there were some amount of propellers that did not generate. This could be due to the ψ distribution needing to be changed for those alternate distributions or those distributions not working outright. While the reference line for screw propellers is simple, a toroidal propeller's standardized reference line will require more insight to develop. To be able to build an equivalent toroidal propeller from a screw propeller, the reference line will be essential, and as shown in this work, the large variations in efficiency show that these parameters cannot be selected randomly.

The outcomes of this paper show that it is unlikely that a TORUS propeller would be a better choice than the standard screw propeller. However, if there was a scenario where the efficiency does not matter and the only requirement is a higher thrust while maintaining the same package size, the TORUS propellers would be a solution.

A Github

The core of all the geometry generation code can be found within the **Geometry** repository on Github, found here:

<https://github.com/TeamTorus/Geometry>

This repository contains the following key entry points and files:

- `/pointcloud_approach/MatLab Point Cloud/0000BladePoints.txt`
The raw data for the various airfoils along the blade.
- `/pointcloud_approach/MatLab Point Cloud/0000BladeParameters.txt`
The raw data for the configuration of blade macroscopic parameters.
- `/pointcloud_approach/MatLab Point Cloud/ToroidalCreator.m`
Primary MATLAB script to take the raw data and produce a full pointcloud file capable of being used with the Fusion API to generate geometries.
- `/pointcloud_approach/Fusion 360 API Propeller Creator/`
The Autodesk Fusion 360 API and integrations to take the outputs of the pointcloud approach and generate full CAD models from them.
- `/pointcloud_approach/Propellers/`
CAD files for the final toroidal propellers generated, as well as the spreadsheets used to tabulate and modify their parameters.
- `/parametric-eqns/config/toroidal.py`
Input parameters for the purely-geometric Frenet-Serret approach to generating toroidal propellers.
- `/parametric-eqns/nurbs/gen.py`
Code to generate the NURBS spline for the blade centerline in the geometric Frenet-Serret approach.
- `/parametric-eqns/toroidal/prop.py`
Primary file to generate STEP CADs using the geometric Frenet-Serret approach.
- `/parametric-eqns/manual/fit.ipynb`
Jupyter notebook file that provides sliders to modify and manipulate parameters in real time to see how the CAD changes using the geometric Frenet-Serret approach.

The simulation setup files for STAR-CCM to run the CFDs used in this paper can be found in the following repository:

<https://github.com/TeamTorus/toroidal-cfd-sims/tree/main>

B Educational Video

As we previously discussed, this project serves to actively address the environmental issues tied to global greenhouse gas emissions tied to the maritime shipping industry. However, in order to effectively “do good” we have a responsibility to move beyond research and design to increasing the accessibility of cutting-edge science. To fulfill this, we published a high-school level science education video regarding this project [31]. While we aim to prioritize a high school aged audience, we additionally seek to encourage citizen science by engaging non-expert adults. In recent years, educational videos on the publicly accessible YouTube platform have grown in both popularity and success, enhancing learning by increasing student motivation and intrinsic satisfaction with their learning [32].

Something that is frequently overlooked in STEM education, including at our own university, is the approach of exploring STEM with communal goals rather than with individualistic intentions. Recent studies have demonstrated that starting at a younger age, beginning STEM education with these shared aims, like collectively addressing the maritime shipping greenhouse gas emission issue helps improve STEM learning perseverance [33]. Simultaneously, improving access to educational materials with the goal of global collective action to mitigate anthropogenic climate change intensity should improve the sense of belonging and academic hope of students from more marginalized communities [34]. Fundamentally, while the video is structured around explaining the fundamentals of screw and toroidal propeller geometry, CFD, optimization, and ultimately our results, the guiding thread will be one encouraging younger students to pursue STEM pathways. Ultimately, we aim to popularize our results in order to not only move forward with toroidal propeller studies towards a real material use but perhaps more importantly, to take on science education to encourage a new generation of scientists to take up this work.

References

- [1] *Shipping data unctad releases new seaborne trade statistics*, accessed 2025-12-04, Apr. 2025. [Online]. Available: <https://unctad.org/news/shipping-data-unctad-releases-new-seaborne-trade-statistics>.
- [2] *International shipping*, accessed 2025-12-04, 2023. [Online]. Available: <https://www.iea.org/energy-system/transport/international-shipping>.
- [3] F. Herbert and F. Caupin, “Cavitation in water: A review,” *Comptes Rendus Physique*, vol. 7, pp. 1000–1017, 2006. DOI: 10.1016/j.crhy.2006.10.015.
- [4] J. Kerwin, “Marine propellers,” *Annual Review of Fluid Mechanics*, vol. 18, 1986. DOI: 10.1146/annurev.fl.18.010186.002055.
- [5] W. G. Neal, “A new patent screw propeller,” *Marine Engineering, Shipbuilding, Steam Navigation Electrical Engineering*, vol. 13, pp. 241–242, 1891. DOI: needed.
- [6] C. Myers, “Screw propeller,” US467322A, accessed 2025-12-04, Jan. 19, 1892. [Online]. Available: <https://patents.google.com/patent/US467322A/en>.
- [7] O. A. Wirkkala, “Marine propeller,” US2978040A, accessed 2025-12-04, 1958. [Online]. Available: <https://patents.google.com/patent/US2978040A/en>.
- [8] W. G. Walker, “Experiments of the arrangement of the surface of a screw-propeller,” in *Institution of Mechanical Engineers*, vol. 46, Oct. 1892, pp. 547–551.
- [9] H. Haussmann, “Marine propeller,” Patent US547210A, Oct. 1, 1895. [Online]. Available: <https://patents.google.com/patent/US547210A>.
- [10] G. C. Sharrow, “Propeller,” US9926058B2, accessed 2025-12-04, 2018. [Online]. Available: <https://patents.google.com/patent/US9926058B2/en>.
- [11] M. L. Laboratory, *Innovation highlight: Toroidal propeller*, accessed 2025-12-04, Feb. 2023. [Online]. Available: https://www.ll.mit.edu/sites/default/files/other/doc/2023-02/TV0%7B%5C_%7DTechnology%7B%5C_%7DHighlight%7B%5C_%7D41%7B%5C_%7DToroidal%7B%5C_%7DPropeller.pdf.
- [12] I. H. Abbott, A. E. Von Doenhoff, and L. J. Stivers, “Summary of airfoil data,” National Advisory Committee for Aeronautics (NACA), Tech. Rep. NACA–TR–824, 1945. [Online]. Available: <https://ntrs.nasa.gov/api/citations/19930090976/downloads/19930090976.pdf>.
- [13] T. Brockett, “Minimum pressure envelopes for modified naca-66 sections with naca a = 0.8 camber and buships type 1 and type 2 sections,” David Taylor Model Basin, Tech. Rep. DTMB 1780, 1966.
- [14] V. Bertram, “Propellers,” in *Practical Ship Hydrodynamics*, accessed 2025-12-04, Elsevier, 2012. [Online]. Available: <https://www.sciencedirect.com/science/chapter/monograph/abs/pii/B9780080971506100028?via%3Dihub>.

- [15] L. Ye, C. Wa, C. Su, and C. Gu, “Mathematical expression method for geometric shape of toroidal propeller,” *Chinese Journal of Ship Research*, vol. 19, no. 3, pp. 224–233, Jun. 2024.
- [16] *Propeller Geometry: Terms and Definitions*, Michigan Wheel Engineering, Federal Propellers, Grand Rapids, MI, USA, 2000.
- [17] *Propeller propulsion*, accessed 2025-12-04. [Online]. Available: <https://www.grc.nasa.gov/WWW/K-12/airplane/propeller.html>.
- [18] A. Ishiodu et al., “Design procedure of 4-bladed propeller,” *West African Journal of Industrial and Academic Research*, vol. 8, no. 1, pp. 13–24, 2013.
- [19] L. Birk, *Fundamentals of Ship Hydrodynamics: Fluid Mechanics, Ship Resistance and Propulsion*. John Wiley & Sons, 2019, pp. 428–429.
- [20] T. Sebastian and C. Strem, “Toroidal propeller,” US10836466B2, accessed 2025-12-04, Nov. 17, 2020. [Online]. Available: <https://patents.google.com/patent/US10836466B2/en>.
- [21] G. Ion and I. Simion, “Performance of 3d printed conventional and toroidal propeller for small multirotor drones,” *Journal of Industrial Design and Engineering Graphics*, vol. 18, no. 1, pp. 27–32, Jul. 2023.
- [22] S. Flores and S. Leandro, *Development of improvements in uas for difficult access environments*, accessed 2025-12-04, 2023. [Online]. Available: <https://upcommons.upc.edu/handle/2117/391675>.
- [23] A. Schulz, *Development of a low noise drone propeller using cfd simulations*, accessed 2025-12-04, 2023. [Online]. Available: <http://lup.lub.lu.se/student-papers/record/9131449>.
- [24] S. D. Jessup, “Measurement of multiple blade rate unsteady propeller forces,” David Taylor Research Center, Tech. Rep. DTRC-90/015, 1990.
- [25] A. Serani, T. P. Scholcz, and V. Vanzi, “A scoping review on simulation-based design optimization in marine engineering: Trends, best practices, and gaps,” *Archives of Computational Methods in Engineering*, vol. 31, pp. 4709–4737, 2024. DOI: 10.1007/s11831-024-10127-1.
- [26] H. K. Versteeg and W. Malalasekera, *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*, 2nd. Harlow, England: Pearson Education, 2007, ISBN 978-0-13-127498-3.
- [27] Cornell University. “A hands-on introduction to engineering simulations,” EdX. [Online]. Available: <https://www.edx.org/learn/engineering/cornell-university-a-hands-on-introduction-to-engineering-simulations>.
- [28] D. C. Wilcox, *Turbulence Modeling for CFD*, 3rd. DCW Industries, 2023, ch. 1.
- [29] The ManchesterCFD Group. “All there is to know about different mesh types in CFD!” manchestercfd. [Online]. Available: <https://www.manchestercfd.co.uk/post/all-there-is-to-know-about-different-mesh-types-in-cfd>.

- [30] S. B. Denny, "Cavitation and open-water performance tests of a series of propellers designed by lifting-surface methods," Naval Ship Research and Development Center, Tech. Rep. NSRDC 2878, 1968.
- [31] Daniela Colombi, *Team TORUS: Development of a Toroidal Propeller for Marine Applications*, YouTube, [Online Video]. Available: <https://youtu.be/4Vqm494M3r8>. Accessed: May 1, 2026, May 2026.
- [32] D. Pattier, "Science on youtube: Successful edutubers," *International Technology Science and Society Review*, vol. 10, no. 1, pp. 1–15, 2021. DOI: 10.37467/gka-revtechno.v10.2696.
- [33] M. Nalipay, B. Huang, M. Jong, C. Chai, and R. King, "Promoting stem learning perseverance through recognizing communal goals: Understanding the impact of empathy and citizenship," *International Journal of STEM Education*, vol. 11, no. 17, 2024. DOI: 10.1186/s40594-024-00471-w.
- [34] M. P. M.J. Hansen and L. White, "The importance of stem sense of belonging and academic hope in enhancing persistence for low-income, underrepresented stem students," *Journal of STEM Education Research*, vol. 7, pp. 155–180, 2023. DOI: 10.1007/s41979-023-00096-8.