Computational Analysis of Bio-Inspired Transformations on Symmetrical Airfoils

Tejas Waghode[1,2] and Victor Maldonado[1]

[1]Mechanical Engineering, Texas Tech University, 2500 Broadway, Lubbock, 79409, TX, USA.

[2]Jordan & Skala Engineers, 6201 W Plano Pkwy Ste 250, Plano, 75093, TX, USA.

Contributing authors: victor.maldonado@ttu.edu;

Abstract

This computational study delves into the potential of the Fibonacci Sequence and the Golden Ratio in enhancing the aerodynamic efficiency of symmetrical airfoils. The NACA 0012 airfoil—a well-known symmetrical profile—is used as the baseline for creating two novel airfoil geometries; one for each mathematical transformation. These transformations were scripted in Python and ANSYS Workbench with SPEOS is used for performing 2D CFD simulations on the baseline and modified airfoils. The $\boldsymbol{k-\omega}$ ***SST*** (Shear Stress Transport) turbulence model is used in ANSYS to achieve airflow simulation at a Reynolds number of 7.4 million. The lift-to-drag ratio values of these airfoils are calculated at angles of attack ranging from 0 to 15 degrees and compared to determine efficiency improvements. The results showed efficiency improvement of both modified airfoils at certain angles of attack.

Keywords: Aerodynamics, Symmetrical Airfoils, Fibonacci Sequence, Golden Ratio, Computational Fluid Dynamics, ANSYS

1 Nomenclature

Table 1 Nomenclature

Symbol	=	Description
α	=	Angle of attack
L	=	Lift Force
C_L	=	Lift Coefficient
D	=	Drag Force
C_D	=	Drag Coefficient
L/D	=	Lift-to-drag ratio
ν	=	Kinematic Viscosity
U_∞	=	Free-stream velocity
ρ	=	Air Density
k	=	Turbulent Kinetic Energy
ω	=	Specific Dissipation Rate
P_k	=	Production of Turbulence
Re	=	Reynold's Number (7.4×10^6)
CFD	=	Computational Fluid Dynamics
$\varnothing$	=	Golden ratio ($\varnothing = 1.618$)
$NACA$	=	National Advisory Committee for Aeronautics

2 Introduction

Humans have often taken inspiration from natural processes to solve engineering problems. This phenomenon is called biomimicry [1, 2]. The Fibonacci Sequence and the Golden Ratio are mathematical concepts frequently appearing in natural processes; hence biomimicry is a term usually paired with them [3–6]. Although the significance of these concepts in nature is exaggerated [7, 8], researchers have been able to, in a few successful cases, incorporate these principles to engineer efficient solutions [1, 9–11]. These bio-inspired principles have been successfully implemented in the following engineering endeavors: designing high-aspect ratio aircraft wings [12], enhancing microfluidics and micro-channel designs [13, 14], micro-power energy harvesting [15], RFID antenna designs [16], improving Helmholtz resonators for noise control [17], photonics and plasmonics [18–21], and robotic systems [22].

The Fibonacci sequence is simply a chain of numbers, where each number is the sum of the previous two numbers. This sequence can be found in the arrangement of pineapple fruitlets, cauliflower heads, pinecone heads, and in flower petals. As the chain of numbers in the Fibonacci Sequence progresses, the ratio between consecutive terms approaches an irrational number of 1.618. This irrational number is the Golden Ratio, denoted by $\varnothing$. The Golden Ratio can be observed in the inner and outer surfaces of seashells, and even in the spirals of galaxies [18, 20, 21, 23–25].

Researchers have been successful in increasing the aerodynamic efficiency of different bodies using the Fibonacci Sequence and the Golden Ratio. A study by Damota et al. achieves a 14% increase in power generation of a vertical axis wind turbine (VAWT), by relying on a Fibonacci-Spiral based blade profile instead of a Savonius type [26]. Similarly, Patil et al. created a novel horizontal axis wind turbine (HAWT) based on the Fibonacci-spiral which displays higher resistance to wind turbulence [27]. Another study by Maldonado develops an efficient and quiet novel axial fan using the Fibonacci Spiral, which reduces average noise by 3.4% and can achieve higher RPM at low torque consumption [28]. Herraprastanti et al. used the Fibonacci Series to create a novel wind turbine called the Archimedes Wind Turbine [29]. This novel wind turbine shows potential of improved structural stability and a higher energy capture, however, unstable values in the simulation requires further research. Su et al. combined the Golden Ratio and the Kutta-Zhukovsky transformation to modify a traditional Clark-Y airfoil [30]. This design demonstrates lower stalling speed and better climbing performance.

The above examples show the potential of these biomimetic mathematical principles in aerodynamics. The current study utilizes a symmetrical NACA 0012 airfoil to develop two novel airfoils: one using the Fibonacci Sequence and the other applying the Golden Ratio. In contrast to existing studies, this research paper purposefully avoids introducing camber or other asymmetric features to the airfoil and focuses on analyzing the effect of each mathematical principle to a symmetrical airfoil. The primary objective is to explain from a fluid perspective how the minor transformations of the baseline airfoil surfaces lead to enhanced aerodynamic efficiency for the specific Reynolds number and angles of attack tested.

3 Methodology

The Fibonacci Sequence is a recursive numerical sequence discovered by and named after the Italian mathematician Leonardo Fibonacci [31]. Fibonacci published his findings in a book titled "Liber Abaci" in 1202. In this sequence, the numbers progress continuously, where the current number is the sum of the two numbers following it. Using this definition, the sequence is represented mathematically as follows:

$$F_n = F_{n-1} + F_{n-2} \tag{1}$$

Using the above formula for the Fibonacci Sequence, the numerical sequence will unfold as follows: 1, 1, 2, 3, 5, 8, 13, 21, ...

The above sequence goes on infinitely and as the variable increases, the ratio between successive terms approaches an irrational number equal to 1.618. The mathematical representation of this irrational number is as follows:

$$\varnothing = \frac{F_{n+1}}{F_n} = \frac{1+\sqrt{5}}{2} = 1.618033989\ldots \approx 1.618 \tag{2}$$

This irrational number that approaches 1.618 is called the Golden Ratio and is denoted by the symbol '$\varnothing$'. Both these mathematical principles are found in nature, e.g., the growth pattern of leaves and the arrangement of seeds in sunflowers.

To perform the geometrical transformation, two separate Python scripts for each principle were written. The Python script for the Fibonacci Sequence transformation is named 'fibonacci_builder', while the code for the Golden Ratio transformation is named 'golden_ratio_builder' included in this article as an appendix. The codes are user-friendly and can be used on any symmetrical airfoil as described. The user inputs an appropriate .DAT file consisting of x-y coordinates of the desired symmetrical airfoil into the Python code. The Python code is written to work only on the two-dimensional coordinates of the airfoil and will not accept three-dimensional coordinates. For this study, the .DAT file for the coordinates of the NACA 0012 are acquired from a trusted and widely used source [32]. Upon reading the .DAT file, the Python code transforms the baseline symmetrical airfoil by modifying the vertical coordinate, y of the airfoil surface according to the desired mathematical principle. In the output, the code provides a new .DAT file with the modified airfoil geometry and provides a graphical comparison plot between the baseline and modified airfoil.

For this study, the number of coordinates of the input baseline NACA 0012 airfoil is 130. Both codes also consist of interpolation functions which sets the number of coordinates of the modified airfoil to 1000. Using a large coordinate set provides a reliable transformed airfoil model. The modified Fibonacci airfoil converges at all angles of attack. The modified Golden Ratio airfoil, however, is unable to converge at all angles of attack. This indicates that the larger coordinate set of the modified Golden Ratio airfoil introduces complexities during the ANSYS simulation. The interpolation function in the 'golden_ratio_builder' is removed to fix this issue, which enables convergence at all angles of attack.

3.1 Fibonacci Sequence (Python Code 'fibonacci_builder')

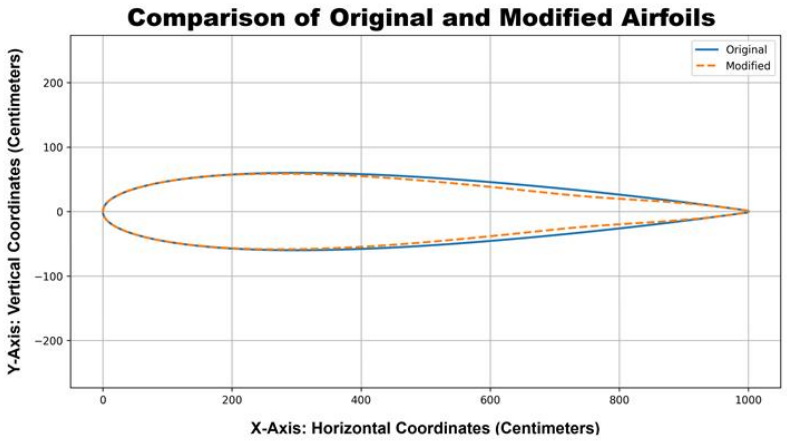

Fig. 1 2D plot of airfoils on Python using .DAT coordinates—FIBONACCI AIRFOIL (orange outline) VS NACA 0012 (blue outline)

The 'fibonacci_builder' is the Python code developed for this study to transform any symmetrical airfoil using the Fibonacci Sequence. The code analyzes the x-y coordinates provided in the .DAT file and modifies the vertical coordinate y of the airfoil surface by using a polynomial-based equation depicting the Fibonacci spiral pattern as follows:

$$y_{\text{new}} = y_{\text{original}} + (A \cdot x^n \cdot (1 - x)) \tag{3}$$

In the above formula, A is the scaling factor which controls the amplitude of the transformation, and 'n' is the power exponent controlling the curvature of the transformation. This code allows the user to use any desired scale and power exponent according to the airfoil or study requirements.

In this study, a scale of -0.085 and a power exponent of 3 is chosen to ensure that the thickness of the modified airfoil is close to the thickness of the baseline airfoil. This enables the removal of any bias for this study so the baseline and modified airfoils can be compared fairly. The scaling factor and power exponent used in the study ensures that the modification blends seamlessly by blending the leading and trailing edges smoothly to avoid discontinuities. To perform numerical operations, plotting, and interpolation—numpy, matplotlib.pyplot, and scipy.interpolate libraries are imported into the code.

3.2 Golden Ratio (Python Code 'golden_ratio_builder')

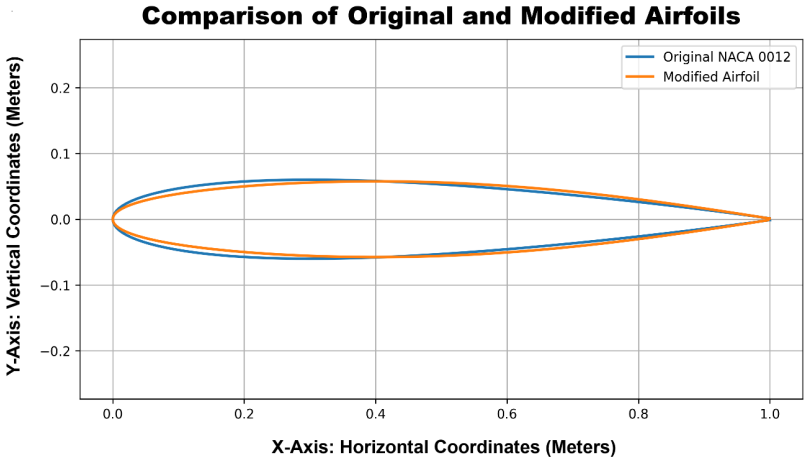

Fig. 2 2D plot of airfoils on Python using .DAT coordinates—GOLDEN RATIO AIRFOIL (orange outline) VS NACA 0012 (blue outline)

The 'golden_ratio_builder' is the Python Code developed for this study to transform any symmetrical airfoil using the Golden Ratio. The vertical coordinate y of the baseline airfoil is modified using a sinusoidal function influenced by the Golden Ratio below:

$$y_{\text{new}} = y_{\text{original}} + (\varnothing_{\text{factor}} \cdot \sin(\pi x) \cdot (\varnothing - x) \cdot x) \tag{4}$$

In the above equation, $\varnothing$ is the Golden Ratio ≈ 0.618, and the $\varnothing_{\text{factor}}$ is a constant controlling the transformation strength. This equation transforms the upper surface of the baseline airfoil and then mirrors the upper surface to maintain perfect symmetry. In this study, a $\varnothing_{\text{factor}}$ of 0.03 is used to ensure that the thickness of the modified airfoil is close to the thickness of the baseline airfoil. This code also uses numpy, matplotlib.pyplot, and scipy.interpolate libraries to perform numerical operations, plotting, and interpolation.

3.3 Settings for ANSYS Workbench with SPEOS

After successfully modifying the baseline airfoil using the desired mathematical principles, the aerodynamic characteristics and efficiency changes are observed by performing 2D CFD simulations in a software called ANSYS Workbench with SPEOS [33]. Initially, a 2D rectangular flow domain with 15 upstream chords, 25 downstream chords, and 20 vertical chords are created. This flow domain is centered around the airfoil that needs to be observed. For the meshing process, a triangular mesh structure is generated. To capture boundary layer effects, inflation layers are added to the triangular mesh near the airfoil surface. The final mesh consists of roughly 220,000 elements with a value of average element quality above 0.9. To ensure appropriate resolution for turbulence modeling, the first cell height is chosen to maintain the constraint of $y^+ = 1.8$.

To accurately observe and compare aerodynamic efficiencies between the modified airfoil and the baseline airfoil, capturing near-wall and observing free-stream flow behavior is essential. To meet these conditions, the $k-\omega$ SST (Shear Stress Transport) turbulence model is used for all simulations. The equations used in the simulations are as follows:

Continuity Equation

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \overrightarrow{v}) = 0 \tag{5}$$

Momentum Equation

$$\frac{\partial (\rho \vec{v})}{\partial t} + \nabla \cdot (\rho \vec{v} \vec{v}) = -\nabla p + \nabla \cdot (\tau) + \rho \vec{g} \tag{6}$$

Turbulent Transport Equations (k and ω)

$$\frac{\partial k}{\partial t} + (\overrightarrow{v} \cdot \nabla k) = P_k - \beta^* k\omega + \nabla \cdot [(\nu + \sigma_k \nu_t) \nabla k] \tag{7}$$

$$\frac{\partial \omega}{\partial t} + (\overrightarrow{v} \cdot \nabla \omega) = \alpha \frac{\omega}{k} P_k - \beta \omega^2 + \nabla \cdot [(\nu + \sigma_\omega \nu_t) \nabla \omega] + 2(1 - F_1) \frac{\sigma_{\omega 2}}{\omega} \nabla k \cdot \nabla \omega \tag{8}$$

The flow is simulated at a Reynolds number of 7.4 million, density of 1.229 kg/m^3, temperature of 288.16 K, free stream velocity of 52.08 m/s and a viscosity of 8.68e-06 kg/ms. The aerodynamic coefficients of the airfoils are computed at angles of attack,

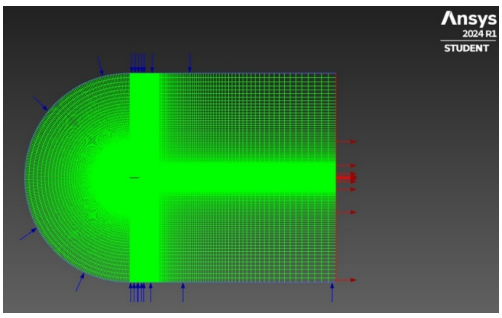

Fig. 3 Flow Domain Meshing using NACA 0012 as the subject airfoil.

from 0° to 15° in 2° increments. For spatial discretization, second-order upwind schemes are employed, and SIMPLE algorithm Is used for pressure-velocity coupling. The convergence of the calculation is dependent on a residual criterion of 10^{-6} and by the stabilization of lift and drag coefficients over 500 iterations.

The turbulence modeling resource for NACA 0012 airfoils by Langley Research Center has been used to source the above values [34]. An initial ANSYS set up is created for the NACA 0012 airfoil and is calibrated using the turbulence modeling resource. This set up is calibrated until the simulations provide the lift and drag coefficients within ± 1% of the accepted published values. Upon calibration, this same procedure is utilized to compute the 2D CFD calculations of both modified airfoils. The lift-to-drag ratio values of the modified airfoils are calculated at a range of angles of attack and are compared with the baseline NACA 0012 airfoil to observe changes to the coefficients. Using this consistent set up provides relevant aerodynamic predictions for comparison and reduces calculation errors.

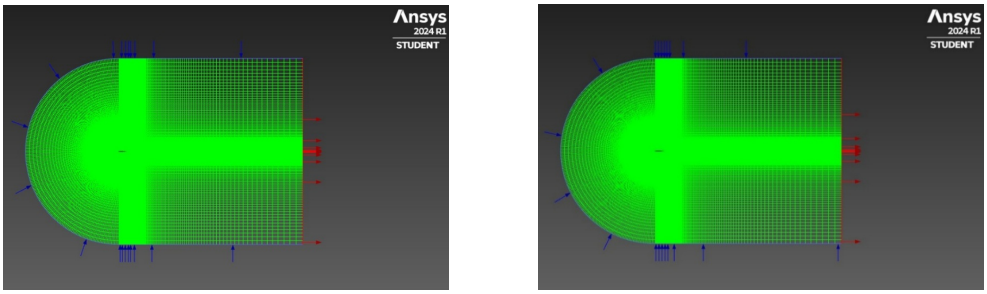

Fig. 4 Flow domain meshing using (left) Fibonacci modified airfoil and (right) Golden Ratio modified airfoil as the subject airfoils.

4 Results and Discussion

The baseline and modified airfoil data from the CFD simulations is tabulated in Table 2 from angles of attack, $\alpha = 0°$ to $15°$. This data is then plotted in Figures 5, 6, 7, and discussed in the following subsections. The pressure contour, velocity contour, the velocity vector, and the turbulent kinetic energy contour of the NACA 0012 airfoil are plotted at $\alpha = 8°$ in Figures 8 and 9 to compare efficiency changes in the modified airfoils.

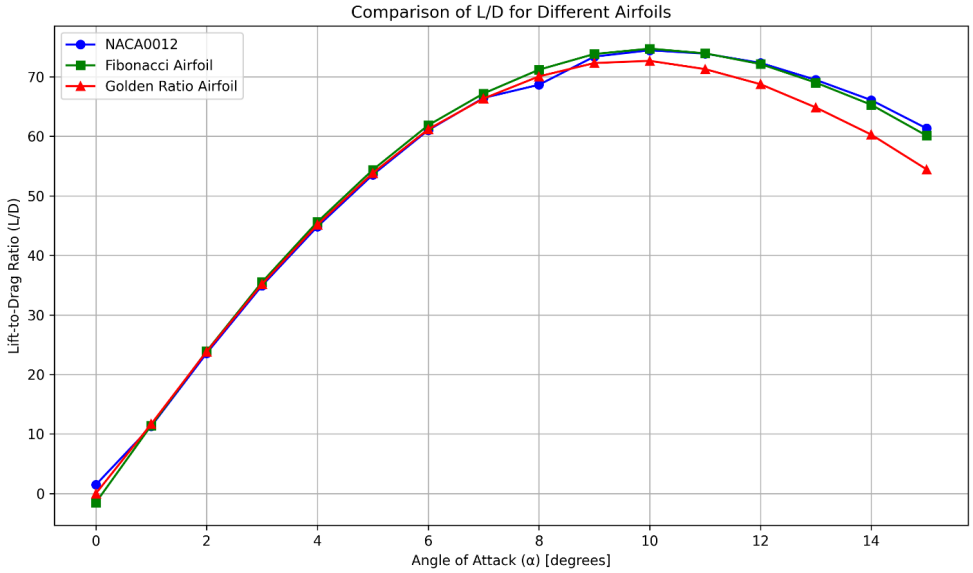

Fig. 5 Line graph of L/D vs Angle of Attack (α) for each airfoil plotted using L/D values from Table 2.

Table 2 Comparison of airfoil performance at Re=7.4 million

	NACA 0012			FIBONACCI				GOLDENRATIO			
Angle of Attack (α)	C_l	C_d	L/D	C_l	C_d	L/D	% change	C_l	C_d	L/D	% change
0°	0.011817	0.0080318	1.471	-0.012051	0.0079324	-1.519	-203.3%	-0.000013619	0.0079188	-0.00172	-100.12%
1°	0.09129	0.0080835	11.29	0.091297	0.0079849	11.43	1.24%	0.093016	0.0079705	11.67	3.37%
2°	0.1936	0.0082225	23.54	0.19417	0.0081255	23.90	1.53%	0.19319	0.0081052	23.84	1.27%
3°	0.29504	0.008454	34.90	0.29678	0.0083599	35.51	1.75%	0.29293	0.0083302	35.16	0.75%
4°	0.39522	0.0088197	44.82	0.39827	0.0087334	45.60	1.74%	0.39216	0.0086892	45.13	0.69%
5°	0.49583	0.0092659	53.50	0.49939	0.0091892	54.35	1.59%	0.49136	0.0091328	53.79	0.54%
6°	0.59654	0.0097789	61.00	0.60069	0.0097156	61.83	1.36%	0.59021	0.0096509	61.15	0.25%
7°	0.69682	0.01049	66.41	0.70166	0.010446	67.17	1.14%	0.6885	0.010381	66.33	-0.12%
8°	0.79627	0.011602	68.62	0.80184	0.011268	71.17	3.72%	0.78551	0.011216	70.03	2.06%
9°	0.89434	0.012191	73.38	0.9009	0.012207	73.80	0.57%	0.88056	0.012183	72.27	-1.51%
10°	0.99035	0.013306	74.45	0.99825	0.013364	74.69	0.32%	0.97282	0.01339	72.64	-2.43%
11°	1.0836	0.01467	73.86	1.0931	0.014789	73.93	0.09%	1.0614	0.014894	71.28	-3.49%
12°	1.1733	0.016227	72.31	1.1846	0.016426	72.10	-0.29%	1.1451	0.016661	68.72	-4.96%
13°	1.2584	0.018118	69.46	1.2715	0.018429	68.99	-0.68%	1.2225	0.01886	64.82	-6.68%
14°	1.3374	0.020245	66.06	1.3521	0.020713	65.28	-1.18%	1.2915	0.02143	60.26	-8.78%
15°	1.4079	0.022964	61.32	1.4239	0.023695	60.10	-1.99%	1.3497	0.024797	54.43	-11.24%

Note: All data measured at Reynolds number 7.4 million. Percentage change calculated relative to NACA 0012 performance.

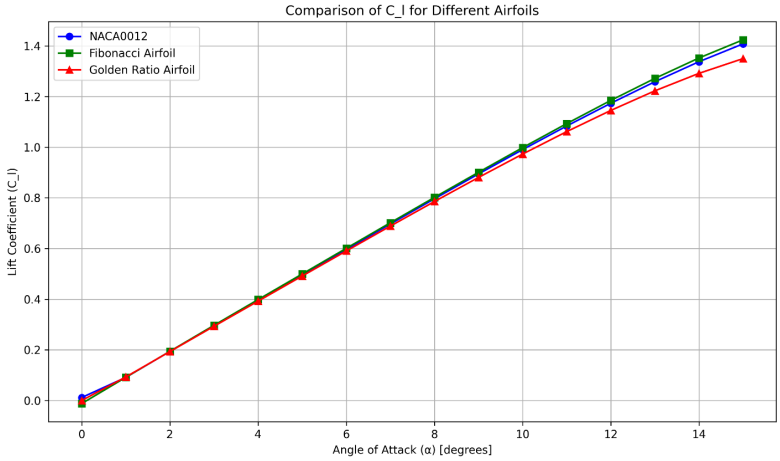

Fig. 6 Line graph of C_l vs Angle of Attack (α) for each airfoil plotted using C_l from Table 2.

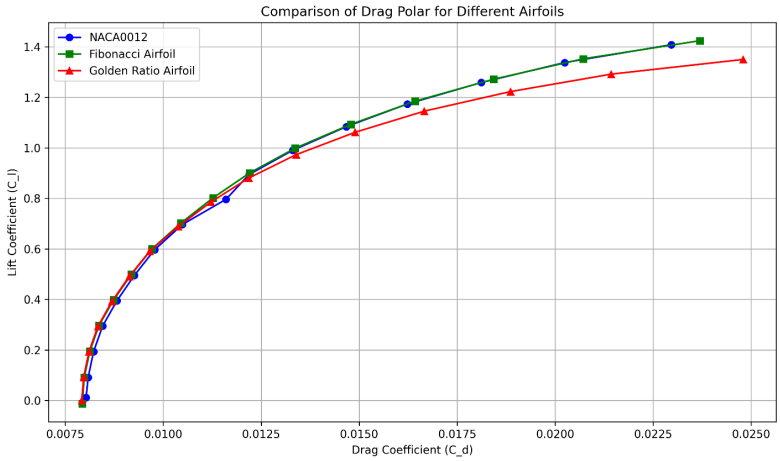

Fig. 7 Line graph of C_l vs C_d for each airfoil plotted using C_l and C_d values from Table 2.

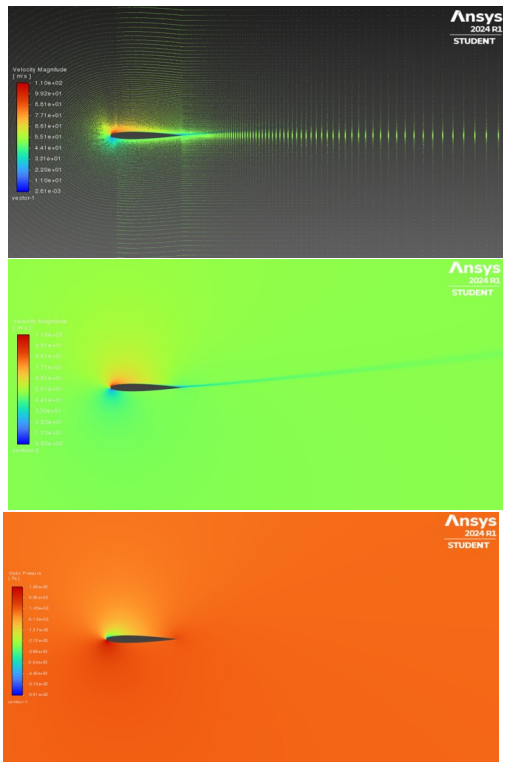

Fig. 8 Velocity vector (top), velocity contour (middle), and pressure contour (bottom) of the NACA 0012 at $\alpha = 8°$ plotted using ANSYS Workbench with SPEOS 2024.

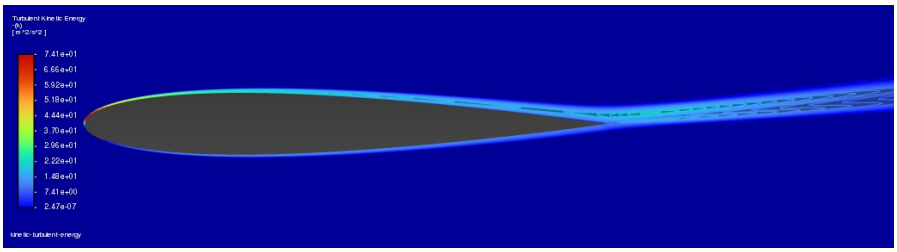

Fig. 9 Turbulent kinetic energy contours of the NACA 0012 at $\alpha = 8°$ plotted using ANSYS Workbench with SPEOS 2024.

4.1 Fibonacci Airfoil

The Fibonacci airfoil with 1,000 coordinates converges at all angles of attack and demonstrates minimally increased L/D values at a majority of angles of attack. The calculated 2D CFD results of this airfoil are compared with the calculated 2D CFD results of the baseline airfoil. There is a 1% - 2% increase in L/D between angles of attack of 1° to 7°. The L/D value peaks at an angle of 8°, showing a 3.68% increase. An increase below 1% in L/D values is calculated from 9° to 11° followed by a small percentage decrease in L/D, at higher angles of attack (12° to 15°).

In order to decouple the effects of C_L and C_D from the L/D plot, C_L is plotted as a function of angle of attack in Figure 6. It can be observed that the Fibonacci airfoil produces very minimally higher values of C_L beyond the stall angle of attack at $\alpha = 10°$, while the golden ratio produces considerably less lift in the stall region up to $\alpha = 15°$. This sharper drop in C_L with the golden ratio airfoil appears to be characteristic of leading edge stall, which occurs with thinner airfoils or those with a smaller leading edge radius. Relative to baseline NACA 0012 airfoil, the golden ratio airfoil satisfies this description while Fibonacci and baseline airfoil appear to have about the same leading edge geometry. The departure in drag between the transformed airfoils and the baseline becomes noticeable at $\alpha = 8°$. There appears to be a small increase in C_D for the baseline airfoil, which explains the dip in the L/D curve at this angle of attack. The minimal improvement in L/D of the Fibonacci airfoil corresponds to moderate angles of attack where airfoils operate under most conditions such as cruise. However, worse performance occurs in the stall and post-stall regime. Further analysis is required for both the angles $\alpha = 8°$ and $15°$ in order to understand how the behavior of velocity and pressure may affect the aerodynamics of the airfoil.

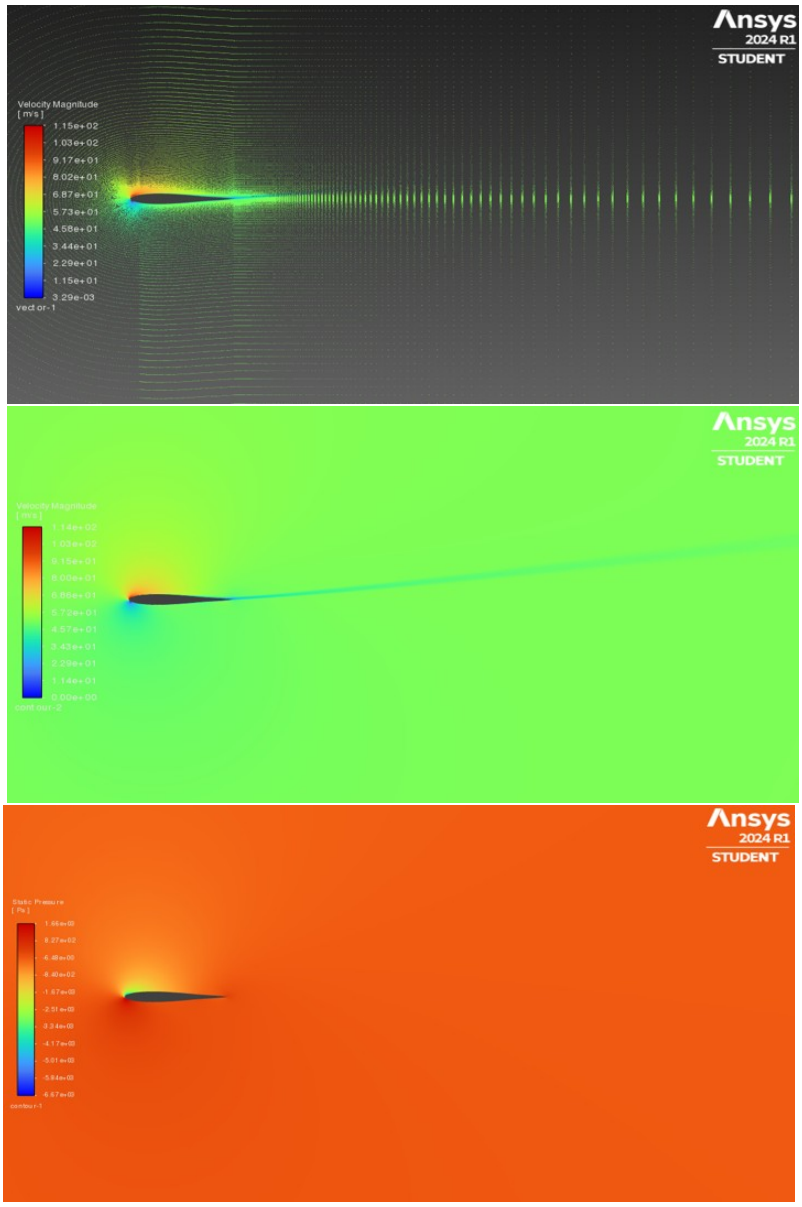

Fig. 10 Velocity vector (top), velocity contour (middle), and pressure contour (bottom) of the modified Fibonacci airfoil at $\alpha = 8°$ (best performing case), plotted using ANSYS Workbench with SPEOS 2024.

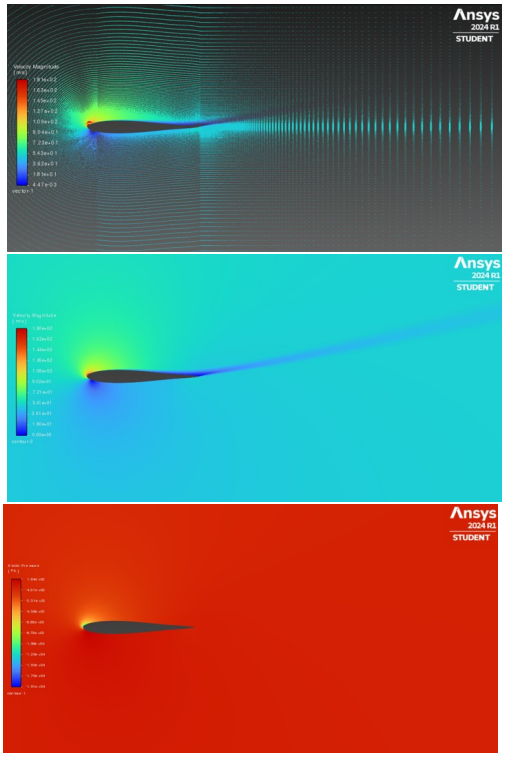

Fig. 11 Velocity vector (top), velocity contour (middle), and pressure contour (bottom) of the modified Fibonacci airfoil at $\alpha = 15°$ (worst case scenario), plotted using ANSYS Workbench with SPEOS 2024.

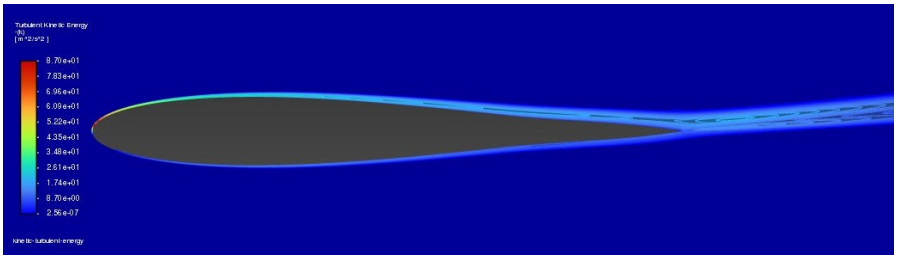

Fig. 12 Turbulent kinetic energy contours of the modified Fibonacci airfoil at $\alpha = 8°$ (best case scenario), plotted using ANSYS Workbench with SPEOS 2024.

4.2 Golden Ratio Airfoil (ANSYS Results)

It is observed that the Golden Ratio airfoil with 1000 number of coordinates does not converge at any angle of attack. This suggests that a larger coordinate set introduces complexities during calculation. To fix this issue, the interpolation features from the Python code are removed.

The Golden Ratio airfoil with a smaller coordinate set converges at all angles of attack. The calculated 2D CFD results of this airfoil are compared with the calculated 2D CFD results of the baseline NACA 0012 airfoil. High L/D values at a 1° angle of attack (3.34% increase) and at a 8° angle of attack (2.04% increase) are observed. Angles of attack from 2° to 6° have a comparatively lower percentage increase in L/D values. All other angles of attack lead to a percentage decrease in L/D values. The hypothesis used to explain the improvements in the Fibonacci airfoil can also be used for the case of the Golden Ratio airfoil. However, the Golden Ratio airfoil does not show any persistent aerodynamic improvements in contrast to the Fibonacci airfoil. The Golden Ratio function may be introducing an unstable surface geometry by introducing sharp, uneven or thick surface contours. It can be hypothesized that the Golden Ratio function in this case introduces a geometry that can promote earlier flow separation. A thick or uneven surface contour can lead to an increased local flow acceleration which can negatively impact flow attachment and pressure drag. Modifying the Python code to further smoothen the surface of the Golden Ratio airfoil can be a possible solution for this issue.

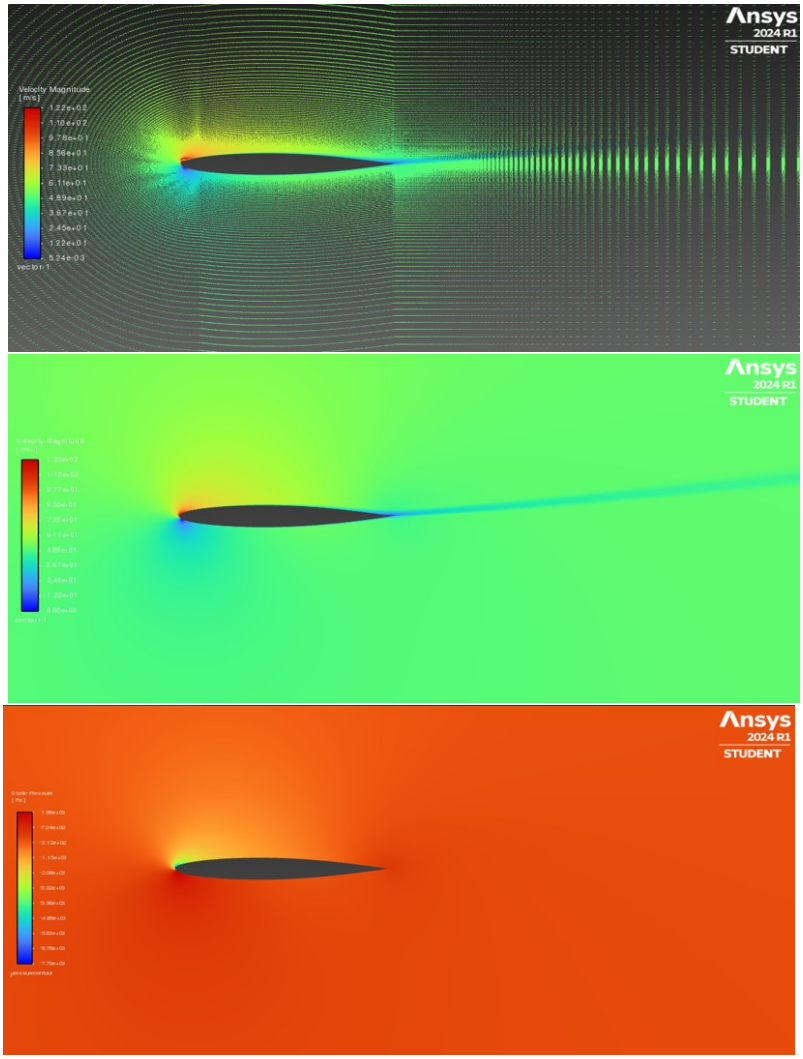

Fig. 13 Velocity vector (top), velocity contour (middle), and pressure contour (bottom) of the modified Golden Ratio airfoil at $\alpha = 8°$ (best performing case), plotted using ANSYS Workbench with SPEOS 2024.

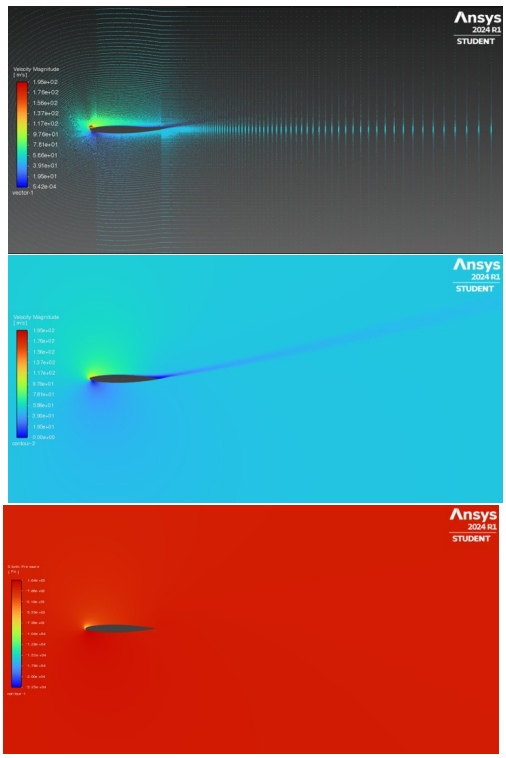

Fig. 14 Velocity vector (top), velocity contour (middle), and pressure contour (bottom) of the modified Golden Ratio airfoil at $\alpha = 15°$ (worst case scenario), plotted using ANSYS Workbench with SPEOS 2024.

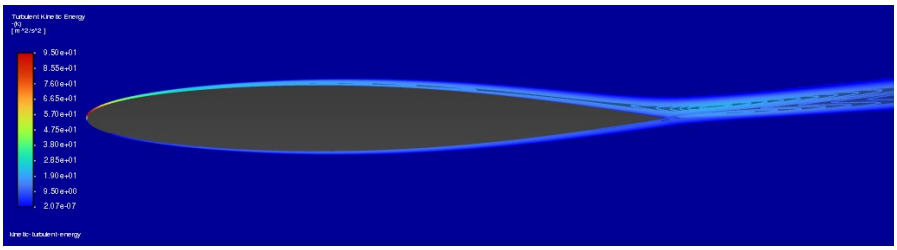

Fig. 15 Turbulent kinetic energy contours of the modified Golden Ratio airfoil at $\alpha = 8°$ (best case scenario), plotted using ANSYS Workbench with SPEOS 2024.

5 Concluding Remarks

This study focuses on the development of two novel symmetrical airfoils inspired by two mathematical principles—The Fibonacci Sequence and the Golden Ratio. These airfoils are designed by using the coordinates from a well-established symmetrical airfoil - NACA 0012. These transformations are performed using Python code and focus on applying the above-mentioned mathematical functions on the coordinates from a .DAT file of any baseline symmetrical airfoil. To calculate and observe the differences in aerodynamic values, ANSYS Workbench with SPEOS (Student Version 2024) is used to perform 2D CFD calculations on the modified and baseline airfoils. The aerodynamic performance of these airfoils is assessed by running the 2D CFD simulations at $0° \leq \alpha \leq 15°$ and $Re = 7.4 \times 10^6$.

A higher performance of the Fibonacci airfoil is observed when compared to the Golden Ratio airfoil and the baseline airfoil (NACA 0012). This performance is measured by calculating the percentage differences in the L/D ratio values at angles of attack from 0° to 15°. The Fibonacci airfoil with a larger set of coordinates can converge at all desired angles of attack, however, this is not the case for the Golden Ratio airfoil with larger set of coordinates. The Golden Ratio airfoil with a smaller set of coordinates is used by removing the interpolation function from the Python code to meet convergence requirements and enable a successful CFD simulation.

The Fibonacci airfoil shows the highest improvement in L/D (increase by 3.68%) at $\alpha = 8°$ when compared to NACA 0012. A moderate to low improvement in L/D (increase by 0.3-2%) at $1° \leq \alpha \leq 11°$, and the gradual decline at $\alpha < 11°$ is noticed in the CFD calculation. These observations portray the Fibonacci airfoil's capabilities to reduce drag and delay flow separation at low angles of attack. Considering the improvements in aerodynamic capabilities at lower angles, the Fibonacci airfoil can be used for maneuvering low-speed aircraft with increased efficiency (e.g. UAVs: Unmanned Aerial Vehicles, and VTOL: Vertical Take-off and landing aircraft. Although the increase in L/D for Fibonacci airfoils is mostly moderate, these small improvements can increase fuel efficiency and enable long range flights in the context of a low-speed aircraft.

The Golden Ratio airfoil revealed its complex structure during the design phase. Reducing the coordinate complexity of the airfoil allowed the convergence of the model during the 2D CFD phase. The Golden Ratio airfoil underperformed significantly at all angles except at $\alpha = 1°$ (3.34% increase in L/D) and at $\alpha = 8°$ (2.04% in L/D). The Golden Ratio airfoil can be beneficial only when used under certain flow conditions and thus can be used in very limited environments. Smoothening the Golden Ratio airfoil surface might lead to some improvements at lower angles of attack. Creating a hybrid airfoil using the positive factors of a Fibonacci airfoil and a Golden Ratio airfoil seems highly beneficial in the context of a low-speed aircraft. Further research needs to be done to fully harness the capabilities of these bio-inspired airfoils. This study can also be applied to automotive and renewable energy fields.

Declarations

Funding: Not applicable.

Conflict of interest/Competing interests: The authors declare that they have no conflict of interest or competing interests.

Ethics approval and consent to participate: Not applicable.

Consent for publication: Not applicable.

Data availability: The data that support the findings of this study are available from the corresponding author upon reasonable request.

Materials availability: Not applicable.

Code availability: The Python codes used in this study are provided in the Appendices.

Author contribution: T.W. developed the Python codes, performed the CFD simulations, and analyzed the data. V.M. conceived the study, supervised the research, and contributed to the interpretation of results. Both authors wrote and reviewed the manuscript.

References

[1] Omidvarnia, F., Sarhadi, A.: Nature-inspired designs in wind energy: A review. Biomimetics **9**(2), 90 (2024) https://doi.org/10.3390/biomimetics9020090

[2] Su, J.N., Zhu, Q.T., Sun, W.X., Zhang, C.: Autorotation of passive microfliers comprising spiral filamentous wings. Phys. Fluids **35**(6) (2023) https://doi.org/10.1063/5.0151713

[3] Akhtaruzzaman, M., Shafie, A.A.: Geometrical substantiation of phi, the golden ratio and the baroque of nature, architecture, design and engineering. Int. J. Arts **1**(1), 1–13 (2012) https://doi.org/10.5923/j.arts.20110101.01

[4] Chavan, A.D., Suryawanshi, C.V.: Correlation of fibonacci sequence and golden ratio with its applications in engineering and science. Int. J. Eng. Manag. Res. **10**(3), 5 (2020) https://doi.org/10.31033/ijemr.10.3.5

[5] Persaud, D., O'Leary, J.P.: Fibonacci series, golden proportions, and the human biology. JP. Fibonacci Series **2**(5) (2015)

[6] Elsheikh, M.E.: Highly flexible wind turbine blades utilizing corrugated surface hinges. Coatings **11**(6), 635 (2021) https://doi.org/10.3390/coatings11060635

[7] Ravindra, C., Rengan, V., B, R.: A systematic review of fibonacci sequence in the human abdominal wall: Facts and reality. Cureus (2022) https://doi.org/10.7759/cureus.33072

[8] Newton, L.J., Wallace, R.D.: X-57 ground dynamics modeling and analysis. (2022). https://doi.org/10.2514/6.2022-3571 . Presented at AIAA Aviation Forum, June 2022

[9] Bagade, P.M., Bagade, P.P., Chaudhari, A., Ranjan, P., Shirke, S., Sedani, C.: Numerical investigation on aerodynamic performance of helical savonius rotor inspired by natural shapes. J. Mines Metals Fuels **71**(10), 22–32 (2023) https://doi.org/10.18311/jmmf/2023/35859

[10] Karim, S., Ibrahim, H., Darus, M.M.: Wing sequence construction from half butterfly method. In: AIP Conference Proceedings, vol. 2472 (2022). https://doi.org/10.1063/5.0092651

[11] Darus, M.M., Ibrahim, H., Karim, S.: Historical review of existing sequences and the representation of the wing sequence. Math. Stat. **11**(3), 87–96 (2023) https://doi.org/10.13189/ms.2023.110303

[12] Guo, L., Tao, J., Wang, C., Zhang, M., Sun, G.: Fuel efficiency optimization of high-aspect-ratio aircraft via variable camber technology considering aeroelasticity. Proc. Inst. Mech. Eng. G J. Aerosp. Eng. **235**(7), 802–815 (2021) https://doi.org/10.1177/0954410020959964

[13] Hsu, C.H., Dang, H.S., Nguyen, T.A.T.: The application of fibonacci sequence and taguchi method for investigating the design parameters on spiral micro-channel. In: Proceedings of the 2016 IEEE International Conference on Applied System Innovation (ICASI) (2016). https://doi.org/10.1109/ICASI.2016.7539784

[14] Hsu, C.H., Jiang, J.C., Dang, H.S., Nguyen, T.A.T., Chang, C.C.: Investigating the design parameters on spiral micro-channel by using fibonacci sequence and taguchi method. Microsyst. Technol. **24**(1), 609–619 (2018) https://doi.org/10.1007/s00542-016-3262-z

[15] Ki, W.H., Lu, Y., Su, F., Tsui, C.Y.: Design and analysis of on-chip charge pumps for micro-power energy harvesting applications. In: 2011 IEEE/IFIP 19th International Conference on VLSI and System-on-Chip (2011). https://doi.org/10.1109/VLSISoC.2011.6081612

[16] Peng, L., Zhang, H., Liu, Z., Wang, H., Zhu, H.: Design of fibonacci sequence rfid antenna using differential evolution. In: 2018 37th Chinese Control Conference (CCC) (2018). https://doi.org/10.23919/ChiCC.2018.8482889

[17] Basirjafari, S.: Innovative solution to enhance the helmholtz resonator sound absorber in low-frequency noise by nature inspiration. J. Environ. Health Sci. Eng. **18**(2), 1451–1464 (2020) https://doi.org/10.1007/s40201-020-00512-w

[18] Fang, L., Fan, W., Bian, G., Wang, R., You, Q., Gu, W., Xia, N., Liao, L., Li, J., Deng, H., Yan, N., Wu, Z.: Sandwich-kernelled agcu nanoclusters with golden ratio geometry and promising photothermal efficiency. Angew. Chem. Int. Ed. **62**(36) (2023) https://doi.org/10.1002/anie.202305604

[19] Barrientos-García, A., Andrade-Lucio, J.A., Sukhoivanov, I.A., Guryev, I., Ruiz-Pinales, J., Rojas-Laguna, R.: Numerical study of photonic crystal fiber with ultra-flattened chromatic dispersion in anomalous and normal dispersion regimes. Optik **126**(13), 1299–1303 (2015) https://doi.org/10.1016/j.ijleo.2015.04.005

[20] Errouas, Y., Ben-Ali, Y., Ouariach, A., Tahri, Z., Bria, D.: Propagation of the electromagnetic waves in one-dimensional asymmetric photonic comb-like structure based on fibonacci chains of grafted resonators. In: Materials Today: Proceedings, vol. 27, pp. 3169–3173 (2020). https://doi.org/10.1016/j.matpr.2020.03.532

[21] Gao, J., Wu, X.F., Zhang, X.G.: Propagation of lamb waves in one dimensional nesting fibonacci superlattices thin plates. In: Applied Mechanics and Materials, vol. 121-126, pp. 306–310 (2012). https://doi.org/10.4028/www.scientific.net/AMM.121-126.306

[22] Rodic, A., Stevanovic, I., Jovanovic, M.D., Urukalo, D.: On building remotely operated underwater robot-explorer with bi-manual poly-articular system. In: Advances in Intelligent Systems and Computing vol. 371, pp. 485–497 (2016). https://doi.org/10.1007/978-3-319-21290-6_48

[23] Patel, S.B.: The fibonacci sequence and numbers. Int. J. Sci. Res. Eng. Manag. **7**(1) (2023) https://doi.org/10.55041/ijsrem17558

[24] Sinha, S.: The fibonacci numbers and its amazing applications. Int. J. Eng. Sci. Invent. **6**(9), 07–13 (2017)

[25] Ashwindran, S.N., Azizuddin, A.A., Oumer, A.N.: Study of $\sqrt{2}$ conjecture in the construction of drag induced wind turbine blade morphology. Evergreen **8**(3), 573–583 (2021) https://doi.org/10.5109/4491649

[26] Damota, J.B., García, J.d.D.R., Casanova, A.C., Miranda, J.T., Caccia, C.G., Galdo, M.I.L.: Analysis of a nature-inspired shape for a vertical axis wind turbine. Appl. Sci. **12**(14), 7018 (2022) https://doi.org/10.3390/app12147018

[27] Patil, Y.: Design, fabrication and analysis of fibonacci spiral horizontal axis wind turbine. Int. J. Aerosp. Mech. Eng. **5**(2), 24–29 (2018)

[28] Maldonado, H.A.Y.: Design, study, and comparison of a new axial fan using the fibonacci spiral and a classic axial fan. Jordan J. Mech. Ind. Eng. **18**(1), 57–67 (2024) https://doi.org/10.59038/jjmie/180106

[29] Herraprastanti, E.H., Saputro, W.A.: Simulation of opening angle of archimedes wind turbine design based on the fibonacci series. Int. J. Eng. Sci. Inf. Technol. **2**(1), 24–29 (2021) https://doi.org/10.52088/ijesty.v2i1.192

[30] Su, C.-C., Lee, W.-S., Ho, C.-W.: Design and Aerodynamic Characteristics of

Golden Ratio Airfoils. Unpublished manuscript or pre-print (n.d.)

[31] Sivaraman, R.: Exploring metallic ratios. Math. Stat. **8**(4), 149–154 (2020) https://doi.org/10.13189/ms.2020.080403

[32] NACA 0012 airfoil (n0012-il). UIUC Airfoil Coordinates Database, Airfoil Tools (n.d.)

[33] ANSYS, Inc.: ANSYS Workbench with SPEOS (Student Version 2024). Version 2024 (2024)

[34] Rumsey, C.: 2d naca 0012 airfoil validation (turbulence modeling resource). Technical report, NASA (Sep 2022). https://turbmodels.larc.nasa.gov/naca0012_val.html

Appendix A Python Code Script (fibonacci_builder)

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.interpolate import interp1d

def fibonacci_curve(x, scale=-0.085, power=3):
    return scale * (x ** power) * (1 - x)

def smooth_step(x, edge0=0, edge1=1):
    x = np.clip((x - edge0) / (edge1 - edge0), 0.0, 1.0)
    return x * x * (3 - 2 * x)

def modify_airfoil(input_file, output_file, chord_length=1000,
    modification_extent=1):
    coords = np.loadtxt(input_file, delimiter=',')
    le_index = np.argmin(coords[:, 0])
    upper_surface = coords[:le_index+1][::-1]
    lower_surface = coords[le_index:]

    upper_interp = interp1d(upper_surface[:, 0], upper_surface[:,
        1], kind='cubic')
    lower_interp = interp1d(lower_surface[:, 0], lower_surface[:,
        1], kind='cubic')

    x_mod = np.linspace(0, 1, 501)
    fib_mod = fibonacci_curve(x_mod)
    transition = smooth_step(x_mod, edge0=modification_extent-0.3,
        edge1=modification_extent)

    upper_mod = upper_interp(x_mod) + fib_mod * (1 - transition)
    lower_mod = lower_interp(x_mod) - fib_mod * (1 - transition)
```

```
414
415     trailing_blend = smooth_step(x_mod, edge0=0.9, edge1=1.0)
416     upper_mod = upper_mod * (1 - trailing_blend) + upper_interp(
417         x_mod) * trailing_blend
418     lower_mod = lower_mod * (1 - trailing_blend) + lower_interp(
419         x_mod) * trailing_blend
420
421     modified_coords = np.vstack((
422         np.column_stack((x_mod, upper_mod))[::-1],
423         np.column_stack((x_mod, lower_mod))[1:]
424     ))
425
426     np.savetxt(output_file, modified_coords, delimiter=',', fmt='
427         %.8f')
428     return modified_coords * chord_length
429
430 def plot_airfoils(original, modified, save_path=None):
431     plt.figure(figsize=(12, 6))
432     plt.plot(original[:, 0], original[:, 1], label='Original',
433         linewidth=2)
434     plt.plot(modified[:, 0], modified[:, 1], label='Modified',
435         linewidth=2, linestyle='--')
436     plt.title('Airfoil Comparison')
437     plt.xlabel('X')
438     plt.ylabel('Y')
439     plt.legend()
440     plt.axis('equal')
441     plt.grid(True)
442     if save_path:
443         plt.savefig(save_path, dpi=300, bbox_inches='tight')
444     plt.show()
445
446 if __name__ == "__main__":
447     input_file = r"Provide path of desired location for saving"
448     output_file = r"Provide path of desired location for saving"
449     image_file = r"Provide path of desired location for saving"
450     chord_length = 1000
451
452     original_coords = np.loadtxt(input_file, delimiter=',') *
453         chord_length
454     modified_coords = modify_airfoil(input_file, output_file,
455         chord_length)
456     plot_airfoils(original_coords, modified_coords, save_path=
457         image_file)
458
459     print(f"Modified airfoil coordinates have been saved to {
460         output_file}")
461     print(f"Comparison plot has been saved to {image_file}")
```

Appendix B Python Code Script (golden_ratio_builder)

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.interpolate import interp1d

x, y = [], []
with open(r"Path to .DAT file of baseline airfoil", "r") as f:
    for line in f.readlines()[1:]:
        values = line.strip().split(',')
        x.append(float(values[0]))
        y.append(float(values[1]))

x, y = np.array(x), np.array(y)
half_length = len(x) // 2
thickness = y[:half_length] - y[half_length:][::-1]
max_thickness = np.max(thickness)
max_thickness_index = np.argmax(thickness)
x_max_thickness = x[max_thickness_index]

print(f"True maximum thickness: {max_thickness:.6f}")
print(f"Location of maximum thickness: x/c = {x_max_thickness:.6f}"
    )

upper_surface = np.column_stack((x[:half_length], y[:half_length]))
lower_surface = np.column_stack((x[half_length:], y[half_length:]))

upper_interp = interp1d(upper_surface[:, 0], upper_surface[:, 1],
    kind='cubic', bounds_error=False, fill_value="extrapolate")
lower_interp = interp1d(lower_surface[:, 0], lower_surface[:, 1],
    kind='cubic', bounds_error=False, fill_value="extrapolate")

x_mod = np.linspace(0, np.max(x) - 1e-6, 500)
upper_y_mod = upper_interp(x_mod)
lower_y_mod = lower_interp(x_mod)

def golden_ratio_transform(x, y, true_max_thickness,
    reduction_factor=0.48, phi_factor=0.03):
    phi = (1 + np.sqrt(5)) / 2
    y_transformed = y + phi_factor * np.sin(np.pi * x) * (phi - x)
        * x
    max_transformed_thickness = np.max(y_transformed) - np.min(
        y_transformed)
    target_thickness = true_max_thickness * reduction_factor
    scale_factor = target_thickness / max_transformed_thickness
    y_transformed *= scale_factor
    return y_transformed

```

```
upper_y_mod_transformed = golden_ratio_transform(x_mod, upper_y_mod
    , max_thickness)
lower_y_mod_transformed = -upper_y_mod_transformed

x_combined = np.concatenate((x_mod, x_mod[::-1]))
y_combined = np.concatenate((upper_y_mod_transformed,
    lower_y_mod_transformed[::-1]))
x_combined[-1] = x_combined[0]
y_combined[-1] = y_combined[0]

with open(r"Path to desired ", "w") as f:
    f.write("Modified NACA 0012 with Golden Ratio\n")
    for xi, yi in zip(x_combined, y_combined):
        f.write(f"{xi:.6f},{yi:.6f}\n")

plt.figure(figsize=(10, 5))
plt.plot(x, y, label='Original NACA 0012', linewidth=2)
plt.plot(x_combined, y_combined, label='Modified Airfoil',
    linewidth=2)
plt.title('Comparison of Original and Modified Airfoils (Golden
    Ratio)')
plt.xlabel('x/c')
plt.ylabel('y/c')
plt.legend()
plt.axis('equal')
plt.grid(True)
plt.savefig(r"Provide path of desired location for saving", dpi
    =300, bbox_inches='tight')
plt.show()

print("Modified airfoil DAT file and comparison plot have been
    saved.")
```