Jurnal Rekayasa Mesin
p-ISSN: 1411-6863, e-ISSN: 2540-7678
Vol. 17, No. 3, Desember **2022**, hal.485-494

# Implementation of the Airfoil parameterization PARSEC Method in Python

**M. Hilman Gumelar Syafei[1*], Ragil Tri Indrawati [2] dan Tanwir Ahmad Farhan[3]**

[1]Teknik Mesin, Jurusan Teknik Mesin, Universitas Negeri Semarang,
Sekaran, Gunung Pati, Semarang, Jawa Tengah, Indonesia 50229
[2]Teknik Mesin, Jurusan Teknik Mesin, Politeknik Negeri Semarang
Jl. Prof. Soedarto, Tembalang, Kec. Tembalang, Kota Semarang, Jawa Tengah, Indonesia 50275
[3] Departemen Teknik Mesin, Teknik Mesin, Universitas Indonesia
Kampus Baru UI, Depok, Jawa Barat, Indonesia 16424
*E-mail: m.hilman@mail.unnes.ac.id

Diajukan: 02-11-2022; Direvisi: 11-12-2022; Dipublikasi: 23-12-2022

**Abstrak**

Airfoil telah mengalami banyak perkembangan dan optimalisasi melalui berbagai cara. Terdapat berbagai teknik yang dapat digunakan untuk memodifikasi bentuk arifoil standar yang ada saat ini. Metode PARSEC merupakan salah satu teknik parameterisasi yang paling banyak digunakan untuk memodifikasi bentuk airfoil. Paper ini menyediakan penjelasan yang komprehensif.mengenai implementasi metode PARSEC pada bahasa pemrograman *python*. Studi ini menggunakan Airfoil standar jenis NACA0012 sebagai referensi untuk mengevaluasi metode PARSEC. Airfoil NACA0012 yang dihasilkan dari metode PARSEC kemudian di komparasikan dengan airfoil asli dari NACA0012. Diperoleh bahwa metode PARSEC dapat menghasilkan bentuk arifoil NACA0012 sangat identik dengan airofil NACA0012 yang asli. Pada rentang sudut serang tertentu, nilai $C_L$ dan $C_D$ dari airfoil yang dihasilkan dari metode PARSEC memiliki nilai $C_L$ dan $C_D$ yang hampir sama dengan nilai $C_L$ dan $C_D$ dari airfoil NACA0012 yang asli. Kedua airfoil, NACA0012 yang asli atapun yang dihasilkan dari metode PARSEC memiliki sudut serang optimum yang sama, yaitu 8°. Nilai $C_L/C_D$ maksimum dari NACA0012 yang asli dengan NACA0012 yang dihasilkan dari metode PARSEC secara berturut-turut adalah 84.92 dan 84.28. Diharapkan bahwa studi ini dapat menjadi petunjuk praktis bagi peneliti dan praktisi dalam menggunakan metode PARSEC untuk berbagai kebutuhan.

**Kata kunci:** Airfoil; NACA; PARSEC; Python

***Abstract***

*Airfoil has been developed and optimized in various ways. Various technique had been used to modify the existing standard airfoil shape. PARSEC method becomes one of the most common parameterization techniques that used to modify the airfoil shape. This paper provides comprehensive explanation of the implementation of PARSEC method in the python programming language.This study uses the standard NACA0012 airofil as a reference to evaluate the PARSEC method. The resulting NACA0012 airfoil from PARSEC method is then compared with the original NACA012 airfoil. It is obtained that the PARSEC method can produce almost identical airfoil geometry to the original NACA0012. Under a certain range of angle of attack variation, both CL and CD value of the PARSEC airfoil have a similar trend to the CL and CD value of the original NACA0012. Both PARSEC airfoil and original NACA0012 have the same optimum α angle of 8°. The maximum CL/CD value of the PARSEC airfoil is 84.92, while the maximum CL/CD value of the original NACA0012 is 84.28. It is expected that this study can be a practical guideline for researchers and engineers for applying the PARSEC method for various purposes.*

***Keywords:*** *Airfoil; NACA; PARSEC; Python*

## 1. Introduction

Airfoil design has been developed and has undergone many changes. The form of the airfoil has been developed for various shapes and has different characteristics. Researchers had started to optimize the airfoil shape through various trial and error experiments in wind tunnel [1]. However, the trial-and-error experiment is costly and ineffective. With the advance development of a computational fluid dynamic simulation (CFD) recently, most of an airfoil optimization study is now conducted numerically and integrated with CFD simulation [1]. In addition, several optimization study are also conducted by using the combined Optimization Algorithm and Machine Learning[2], [3]. Generally, airfoil is optimized to maximize the resulting lift force and minimize the resulting drag force. Hence, the objective function of an airfoil shape optimization is finding the maximum ratio of lift coefficient (Cl) and drag coefficient (Cd) [1]. The optimization study of an airfoil shape is also conducted in various applications, including as aircraft[4], gas turbine[5], wind turbine[2], [6], [7] , tidal turbine[3], and ships[8]. Moreover, the recent optimization study is also conducted with multiple objective function, including weight[2] and safety[4]. Therefore, the optimization of the airfoil shape is very essentials in many engineering problems.

Airfoil geometry with a various shapes are provided in literature. To optimize the existing airfoil geometry, it is required a technique to modify the airfoil shape [9] through airfoil parameterization method. The parameterization intends to modify the existing form of an airfoil. This parameterization method is indispensable for optimizing the shape of an airfoil. Currently, there are numerous methods of airfoil parameterization method[9], including B-Splines [9], discrete method[10], Hicks-Hennes functions [11], CST methods [12], [13], and PARSEC method[14]. Among existing airfoil parameterization methods, PARSEC method becomes the most common parameterization methods used in many study [15].

The PARSEC method have been introduced and explained clearly in [14]. The PARSEC method also has been used for airfoil optimization in several studies. The PARSEC method have been used in optimization study of the NACA0012 airfoil[1]. The optimization is conducted using two different optimization algorithms combined with XFoil simulation[17] to evaluate the airfoil performance[1]. The PARSEC method has also been used in optimization study of NACA0012 airfoil by using Taguchi method for the optimization process[15]. Moreover, the PARSEC method has been utilized in multi-Objective optimization study in[16]. Furthermore, several studies have also compare the PARSEC method with the other airfoil parameterization methods [9], [18]–[21]. Nevertheless, there is no study or literature that explain in comprehensive way how the PARSEC method can be implemented in a programming language. An open-acces source code of the PARSEC method implemented in Python programming langugae is provided in [22]. Unfortunately, it only provided the source code without any brief explanation of how the PARSEC method theory is implemented into the python programming language. Hence, the objective of this study is provided the brief explanation of the implementation of PARSEC method in a programming language. This study will explain the step-by-step of the implementaion of the PARSEC method into the procedural proramming using pytohn programming language. The resulting code then will be tested by comparing the existing NACA0012 airfoil coordinate with the NACA0012 depicted by the PARSEC method.

## 2. Material and Method

### 2.1. PARSEC Method

The PARSEC method defines an airfoil shape with 11 parameters as shown in Figure 1. The definition of each parameter in more detail are provided in Table 1. The airfoil shape can be approximated by a curve of upper part and lower part of the airfoil. Each of the airfoil part is defined by a sixth-order polynomial as shown in Equation (1) and Equation (2).

$$\boldsymbol{Z}_{upper} = \sum_{n=1}^{6} a_{up,n} x^{n-1/2} \quad (1)$$

$$\boldsymbol{Z}_{lower} = \sum_{n=1}^{6} a_{low,n} x^{n-1/2} \quad (2)$$

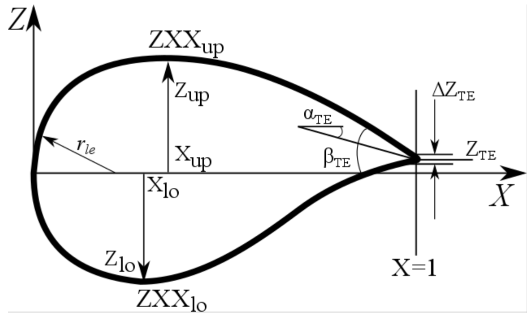


**Figure 1.** The eleven PARSEC parameters define an airfoil geometry [15]

**Table 1.** List of the eleven PARSEC parameters and their definition

| Parameter Index | Symbol | Definitions |
|---|---|---|
| $p_1$ | $R_{LE}$ (m) | Leading Edge Radius |
| $p_2$ | $X_{UP}$ (m) | Upper Crest Abscissa |
| $p_3$ | $Z_{UP}$ (m) | Upper Crest Ordinated |
| $p_4$ | $Z_{XXUP}$ (m) | Upper Crest Curvature |
| $p_5$ | $X_{LO}$ (m) | Lower Crest Abscissa |
| $p_6$ | $Z_{LO}$ (m) | Lower Crest Ordinate |
| $p_7$ | $Z_{XXLO}$ (m) | Lower Crest Curvature |
| $p_8$ | $Z_{TE}$ (m) | Trailing Edge Ordinate |
| $p_9$ | $\Delta Z_{TE}$ (m) | Trailing Edge Thickness |
| $p_{10}$ | $\alpha_{TE}$ (Radiant) | Trailing Edge Direction |
| $p_{11}$ | $\beta_{TE}$ (Radiant) | Trailing Edge Wedge Angle |

The $Z_{upper}$ is the equation for the upper airfoil curve and the $Z_{lower}$ is the lower airfoil curve. The $a_{up}$ and $a_{low}$ are the constant PARSEC matrix for the upper and lower curves successively. So, the Equation (1) and Equation (2) can be re-written in a matrix form as shown in Equation (3).

$$\boldsymbol{Z} = \boldsymbol{AX} \tag{3}$$

Where the matrix $Z$, matrix $A$, and matrix $X$ can be defined as follows:

$$Z = \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_{m-1} \\ Z_m \end{bmatrix}, \quad A = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_5 \\ a_6 \end{bmatrix}, \quad X = \begin{pmatrix} X_1^{0.5} & X_1^{1.5} & \cdots & X_1^{4.5} & X_1^{5.5} \\ X_2^{0.5} & & & & \\ \vdots & & \ddots & & \vdots \\ X_{m-1}^{0.5} & & & & \\ X_m^{0.5} & & \cdots & & X_m^{5.5} \end{pmatrix}$$

The index m refers to the number of the discrete point along the X line across from the leading edge to the trailing edge. The value of all variables inside the matrix A of both $a_{up}$ and $a_{low}$ can be obtained by using the eleven PARSEC parameters by Equation (4) and (5).

$$\boldsymbol{C_{up}}.\boldsymbol{a_{up}} = \boldsymbol{b_{up}} \tag{4}$$

$$\boldsymbol{C_{up}}.\boldsymbol{a_{up}} = \boldsymbol{b_{up}} \tag{5}$$

Where the matrix $C_{up}$ , $C_{low}$, $b_{up}$ and $b_{low}$ are defined as follows:

$$C_{up} = \begin{vmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ p_2^{0.5} & p_2^{1.5} & p_2^{2.5} & p_2^{3.5} & p_2^{4.5} & p_2^{5.5} \\ 0.5 & 1.5 & 2.5 & 3.5 & 4.5 & 5.5 \\ \frac{1}{2}p_2^{-0.5} & \frac{3}{2}p_2^{0.5} & \frac{5}{2}p_2^{1.5} & \frac{7}{2}p_2^{2.5} & \frac{9}{2}p_2^{3.5} & \frac{11}{2}p_2^{4.5} \\ -\frac{1}{4}p_2^{-1.5} & \frac{3}{4}p_2^{-0.5} & \frac{15}{4}p_2^{0.5} & \frac{35}{4}p_2^{1.5} & \frac{63}{4}p_2^{2.5} & \frac{99}{4}p_2^{3.5} \\ 1 & 0 & 0 & 0 & 0 & 0 \end{vmatrix}$$

$$C_{low} = \begin{vmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ p_5^{0.5} & p_5^{1.5} & p_5^{2.5} & p_5^{3.5} & p_5^{4.5} & p_5^{5.5} \\ 0.5 & 1.5 & 2.5 & 3.5 & 4.5 & 5.5 \\ \frac{1}{2}p_5^{-0.5} & \frac{3}{2}p_5^{0.5} & \frac{5}{2}p_5^{1.5} & \frac{7}{2}p_5^{2.5} & \frac{9}{2}p_5^{3.5} & \frac{11}{2}p_5^{4.5} \\ -\frac{1}{4}p_5^{-1.5} & \frac{3}{4}p_5^{-0.5} & \frac{15}{4}p_5^{0.5} & \frac{35}{4}p_5^{1.5} & \frac{63}{4}p_5^{2.5} & \frac{99}{4}p_5^{3.5} \\ 1 & 0 & 0 & 0 & 0 & 0 \end{vmatrix}$$

$$b_{up} = \begin{bmatrix} p_8 + p_9/2 \\ p_3 \\ tan(p_{10} - p_{11}/2) \\ 0 \\ p_4 \\ \sqrt{2p_1} \end{bmatrix}, \quad b_{low} = \begin{bmatrix} p_8 - p_9/2 \\ p_6 \\ tan(p_{10} + p_{11}/2) \\ 0 \\ p_7 \\ -\sqrt{2p_1} \end{bmatrix}$$

Where the index of the eleven PARSEC parameters from $p_1$ to $p_{11}$ may refer to the Table 1. The result of the PARSEC method is a set of points that creates a coordinate of an airfoil.

## 2.2. PARSEC Method Procedure

The PARSEC methods can be used to build a particular shape of airfoil geometry by several steps. This section will explain the step-by-step how to use the PARSEC method appropriately. The procedure of using the PARSEC method is depicted in Figure 2.

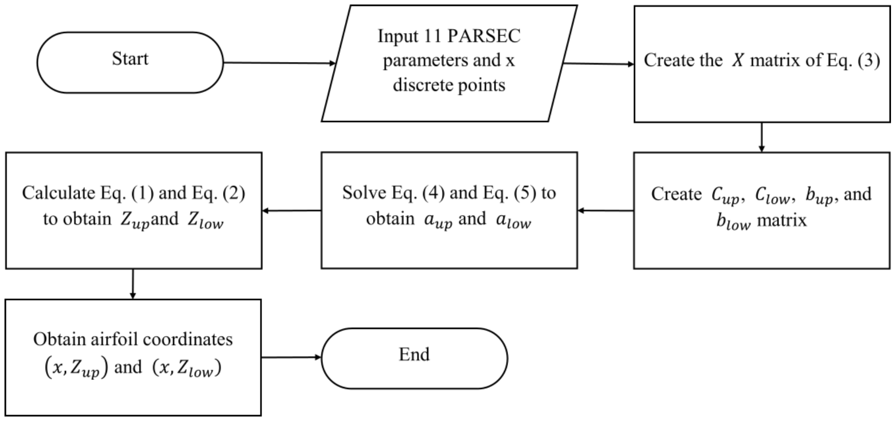


**Figure 2**. Procedure of the PARSEC method

The PARSEC method begins by determining the eleven PARSEC parameters value and the number of x discrete points. Then, the X matrix of Equation (3) can be created. Afterward, the matrix of $c_{up}$, $c_{low}$, $b_{up}$, and $b_{low}$ can be created. Then, the value of $a_{up}$ and $a_{low}$ can be obtained by solving the matrix equations of Equation (4) and Equation (5). Later, the $Z_{up}$ and $Z_{low}$ can be obtained through Equation (1) and Equation (2). Therefore, the coordinates of the upper part $(x, Z_{up})$ and the lower part $(x, Z_{low})$ of the airfoil can be plotted to form the airfoil shape. To give a more comprehensive illustrations the NACA0012 will be approximated by using PARSEC method implemented in the python programming language.

2.3. PARSEC Method Implementation in Python

The PARSEC method can be implemented in the Python programming language. This section provides the ways how to implement the PARSEC method in python. In addition, it will be used to approximates the NACA0012 airfoil. The eleven PARAMETERS of the NACA0012 airfoil is provided in Table 2.

**Table 2.** The PARSEC parameters value of NACA0012 airfoil

| **PARSEC Parameter** | **Value** |
|---|---|
| $R_{LE}$ (m) | 0.0155 |
| $X_{UP}$ (m) | 0.29663 |
| $Z_{UP}$ (m) | 0.06002 |
| $Z_{XXUP}$ (m) | -0.4515 |
| $X_{LO}$ (m) | 0.29663 |
| $Z_{LO}$ (m) | -0.06002 |
| $Z_{XXLO}$ (m) | 0.4515 |
| $Z_{TE}$ (m) | 0 |
| $\Delta Z_{TE}$ (m) | 0.0025 |
| $\alpha_{TE}$ (Radiant) | 0 |
| $\beta_{TE}$ (Radiant) | 0.225 |

The python code for the PARSEC method is initialized by calling the NumPy library. This library is capable to define a matrix as an array and perform various matrix operations. Then, each of the eleven PARSEC parameters of the NACA0012 airfoil is defined as well as provided in Table 2. Note that the chord length of the NACA0012 is considered as one meter. Then, the number of x discrete point of the airfoil can be built as an array using NumPy linspace functions. The python code describing this step can be written as shown in Figure 3. The eleven PARSEC parameters can be defined in a single array variable as can be seen in Figure 4.

```
import numpy as np
#=========================================
# Create PARSEC PARAMETER
#=========================================
R_LE       = 0.0155   #p1
X_UP       = 0.29663  #p2
Z_UP       = 0.06002  #p3
Z_XXUP     = -0.4515  #p4
X_LO       = 0.29663  #p5
Z_LO       = -0.06002 #p6
Z_XXLO     = 0.4514   #p7
Z_TE       = 0        #p8
delta_Z_TE = 0.0025   #P9
alfa_TE    = 0        #p10
betha_TE   = 0.225    #p11
#=========================================
# Create the number of x discrete points
#=========================================
x = np.linspace(0,1,100) #100 discrete points
```

**Figure 3.** Define the PARSEC parameters and create an array containing the x discrete points

```
##Creates array of PARSEC parameters
PARSEC_var = np.array([R_LE,X_UP, Z_UP, Z_XXUP, X_LO,Z_LO,Z_XXLO, Z_TE, delta_Z_TE, alfa_TE, betha_TE])
```

**Figure 4.** Create an array containing all PARSEC parameters

Afterward, the X matrix of Equation (3) can be generated using the for-loop iteration statement. The matrix $c_{up}$, $c_{low}$, $b_{up}$, and $b_{low}$ can also be generated using the for-loop iteration statement as shown in Figure 5 and Figure 6.

```
#=========================================
# Create the X, C_up, C_low,b_up,b_low matrix
#=========================================
n=len(x)                        #number of discrete points
x_matrix = np.zeros ([n,6]) #creates nx6 matrix of zero
for i in  range (0,n):
    for j in range (0,6):
        x_matrix[i,j] = (x[i])**(j+0.5)
######Create the C_up array
C_up = np.ones([6,6])           #creates 6x6 matrix contains number 1
for i in  range (1,6):
    C_up[5,i]=0
for i in  range (0,6):
    C_up[1,i] = (PARSEC_var[1]**(0.5+i))
    C_up[2,i] = 0.5+i
    C_up[3,i] = (PARSEC_var[1]**(-0.5+i))
    C_up[4,i] = (PARSEC_var[1]**(-1.5+i))
C_up[3,:] = [0.5, 1.5, 2.5, 3.5, 4.5, 5.5]*C_up[3,:]
C_up[4,:] = [-1/4, 3/4, 15/4, 15/4, 63/4, 99/4]*C_up[4,:]
#####Create the C_low array
C_low = np.ones([6,6])
for i in  range (1,6):
    C_low[5,i]=0
for i in  range (0,6):
    C_low[1,i] = (PARSEC_var[4]**(0.5+i))
    C_low[2,i] = 0.5+i
    C_low[3,i] = (PARSEC_var[4]**(-0.5+i))
    C_low[4,i] = (PARSEC_var[4]**(-1.5+i))
C_low[3,:] = [0.5, 1.5, 2.5, 3.5, 4.5, 5.5]*C_low[3,:]
C_low[4,:] = [-1/4, 3/4, 15/4, 15/4, 63/4, 99/4]*C_low[4,:]
```

**Figure 5.** Create the CUP and CLow Matrix

```
#####Define b_up
b_up = np.zeros (6)
b_up[0] = PARSEC_var[7] +  PARSEC_var[8]/2
b_up[1] =  PARSEC_var[2]
b_up[2] = np.tan( PARSEC_var[9]- PARSEC_var[10]/2)
b_up[3] = 0
b_up[4] =  PARSEC_var[3]
b_up[5] = (2* PARSEC_var[0])**(0.5)
#####define b_low
b_low = np.zeros (6)
b_low[0] = PARSEC_var[7] -  PARSEC_var[8]/2
b_low[1] =  PARSEC_var[5]
b_low[2] = np.tan( PARSEC_var[9] + PARSEC_var[10]/2)
b_low[3] = 0
b_low[4] =  PARSEC_var[6]
b_low[5] = -(2* PARSEC_var[0])**(0.5)
#calculate a_up and a_low
from numpy.linalg import inv,solve
a_up = solve(C_up,b_up)
a_low = solve(C_low,b_low)
```

**Figure 6.** Create the bUP and bLow matrix and solving the algebraic equations to obtain the aUP and aLow

After the matrix $c_{up}$, $c_{low}$, $b_{up}$, and $b_{low}$ have been generated, the matrix $a_{up}$and $a_{low}$ can be obtained by solving Equation (4) and Equation (5). The Equation (4) and Equation (5) can be solved using the linear algebra solve function provided in the Numpy library as shown in Figure 6. Afterward, $Z_{up}$ and $Z_{low}$ can be obtained by solving Equation (1) and Equation (2). The matrix multiplication operation in Equation (1) dan Equation (2) can be performed by using dot functions provided in the NumPy Library as shown in Figure 7. Finally, the coordinate of both upper $(x, Z_{up})$ and lower $(x, Z_{low})$ part of the airfoil can be plotted to form the airfoil geometry. Note that the dimensions of the array $Z_{low}$, $Z_{up}$, and $x$ should be identical. Therefore, the reshape function can be used to ensure that the array $Z_{low}$, $Z_{up}$, and $x$ have the same dimensions as shown in Figure 7. Later, the aerodynamics of the original NACA0012 and the approximated NACA0012 using PARSEC method will be evaluated using XFoil.

```
#calculate Z_up and Z_low
Z_up = np.dot(x_array,a_up)
Z_low = np.dot(x_array,a_low)
z_up = y_up.reshape([n,1])
z_low = y_low.reshape([n,1])
x = x.reshape([n,1])
```

**Figure 7.** Calculate ZUP and ZLow

## 3. Results and Discussion

The NACA0012 is approximated by using PARSEC method through python programming language. The comparison between the original NACA0012 provided in the literature and the NACA0012 approximated using PARSEC method is shown in Figure 8. It shows that the resulting approximation of the NACA0012 geometry using PARSEC method is identical to the original NACA0012.

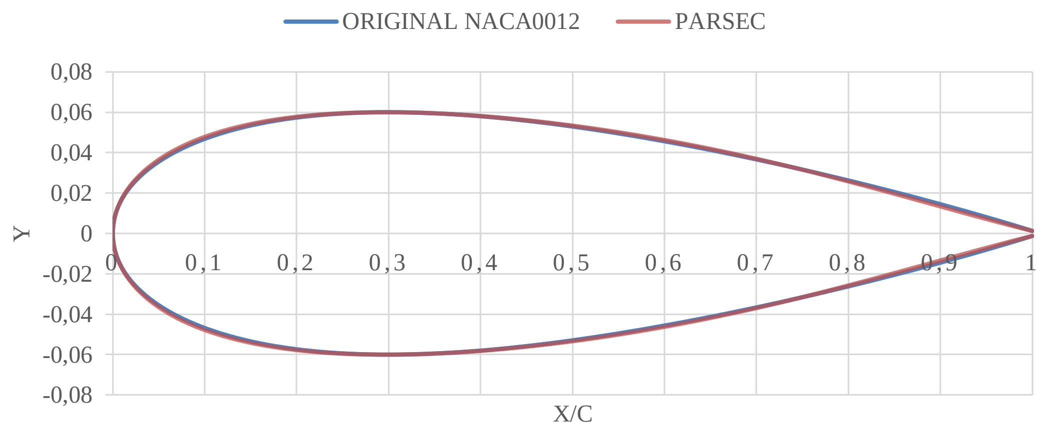


**Figure 8**. Airfoil coordinate plotting graph

To validate the resulting airfoil geometry PARSEC method in this study, aerodynamics performance of both original and PARSEC NACA0012 airfoil will be evaluated. The aerodynamics performance is simulated by using X-Foil open-source software [17].

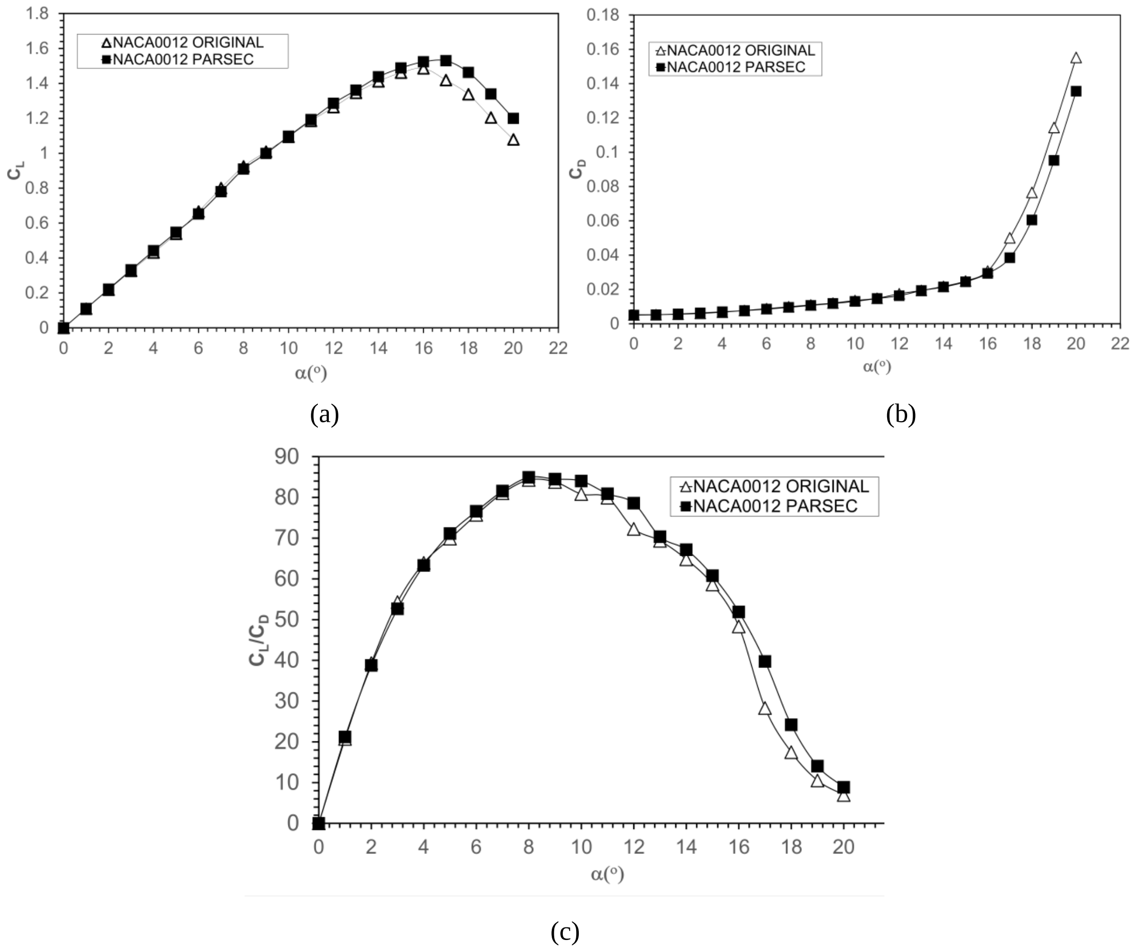


**Figure 9.** Simulation results of (a) lift coefficient, (b) drag coefficient, and (c) ratio of CL/CD of original and PARSEC NACA0021 airofil in particular range of angle of attack (alpha)

The aerodynamics performance is represented by the lift coefficient (CL), drag coefficient (CD), and the ratio of the CL and CD (CL/CD). The value of CL, CD, and CL/CD in a certain range of angle of attack $(\alpha)$ of both original and PARSEC NACA0012 are shown in Figure 4 (a), (b), and (c) respectively. Figure 9(a) and Figure 9(b) shows that both original and PARSEC NACA0012 have almost identical CL and CD at low $\alpha$. When the $\alpha$ exceeds 16°, there is a slight difference of CL and CD between the original and the PARSEC airfoil. The different CL and CD of both airfoils at large $\alpha$ is probably caused by some error value from the XFoil simulation result. Despite, the resulting value of CL and CD of both airfoils have a similar trend.

The most important parameter for evaluating the aerodynamics performance of an airfoil is its CL/CD ratio. Figure 9 (c) shows that both airfoils have a similar trend. Also, both airfoils have the same optimum $\alpha$ angle, that is 8° of $\alpha$ angle. The maximum CL/CD value of the PARSEC NACA0012 airfoil is slightly higher than the original airfoil. The maximum CL/CD value of the PARSEC airfoil and the original airfoil are 84.92 and 84.28 respectively. Therefore, it implies that the PARSEC method in this study is capable to represent the original NACA0012.

## 4. Conclusion

This study provides brief explanation of the implementation of PARSEC method in the python programming language. The NACA0012 airfoil is estimated by using PARSEC method. The resulting airfoil from PARSEC method is then compared with the original NACA012 airfoil. It is obtained that the PARSEC method can produce almost identical airfoil geometry to the original NACA0012. Under a certain range of α angle variation, both CL and CD value of the PARSEC airfoil have a similar trend to the CL and CD value of the original NACA0012. Both PARSEC airfoil and original NACA0012 have the same optimum α angle of 8°. The maximum CL/CD value of the PARSEC airfoil is 84.92, while the maximum CL/CD value of the original NACA0012 is 84.28. It implies that the PARSEC method in this study is sufficiently capable to represent the original NACA0012. It is expected that this study can be a practical guideline for researchers and engineers for applying the PARSEC method for various purposes.

## Reference

[1] M. A. Andira and C. A. Putra, “Comparative Study of Nature-InspiredOptimization Algorithms for Aerodynamics Shape Problem of Subsonic Airfoil,” 2016.

[2] H. M. Lee and O. J. Kwon, “Performance improvement of horizontal axis wind turbines by aerodynamic shape optimization including aeroealstic deformation,” Renew. Energy, vol. 147, pp. 2128–2140, 2020.

[3] L. Wang et al., “A deep learning-based optimization framework of two-dimensional hydrofoils for tidal turbine rotor design,” Energy, vol. 253, p. 124130, 2022.

[4] D. A. I. Jinzhao, L. I. Haoran, Y. ZHANG, and C. Haixin, “Optimization of multi-element airfoil settings considering ice accretion effect,” Chinese J. Aeronaut., 2022.

[5] W. Wang, B. Li, Y. Tan, B. Li, and Y. Shuai, “Multi-objective optimal design of NACA airfoil fin PCHE recuperator for micro-gas turbine systems,” Appl. Therm. Eng., vol. 204, p. 117864, 2022.

[6] S. Sanaye and A. Hassanzadeh, “Multi-objective optimization of airfoil shape for efficiency improvement and noise reduction in small wind turbines,” J. Renew. Sustain. Energy, vol. 6, no. 5, p. 53105, 2014.

[7] F. Grasso, “Usage of numerical optimization in wind turbine airfoil design,” J. Aircr., vol. 48, no. 1, pp. 248–255, 2011.

[8] C. Çelik, D. B. Danışman, S. Khan, and P. Kaklis, "A reduced order data-driven method for resistance prediction and shape optimization of hull vane," Ocean Eng., vol. 235, p. 109406, 2021.

[9] D. A. Masters, N. J. Taylor, T. C. S. Rendall, C. B. Allen, and D. J. Poole, "Geometric comparison of aerofoil shape parameterization methods," AIAA J., vol. 55, no. 5, pp. 1575–1589, 2017.

[10] A. Jameson, "Aerodynamic design via control theory," J. Sci. Comput., vol. 3, no. 3, pp. 233–260, 1988.

[11] J. Reuther, "Aerodynamic shape optimization using control theory," 1996.

[12] B. Kulfan and J. Bussoletti, "' Fundamental' Parameteric Geometry Representations for Aircraft Component Shapes," in 11th AIAA/ISSMO multidisciplinary analysis and optimization conference, 2006, p. 6948.

[13] B. M. Kulfan, "Universal Parametric Geometry Representation Method," J. Aircr., vol. 45, no. 1, pp. 142–158, Jan. 2008.

[14] H. Sobieczky, "Parametric airfoils and wings," in Recent development of aerodynamic design methodologies, Springer, 1999, pp. 71–87.

[15] J. C. Yu and R. Wulandari, "Airfoil aerodynamics optimization under uncertain operating conditions," in Journal of Physics: Conference Series, 2020, vol. 1446, no. 1, p. 12014.

[16] S. Jung, W. Choi, L. S. Martins-Filho, and F. Madeira, "An Implementation of Self-Organizing Maps for Airfoil Design Exploration via Multi-Objective Optimization Technique," J. Aerosp. Technol. Manag., vol. 8, no. 2, pp. 193–202, 2016.

[17] M. Drela, "XFOIL: An analysis and design system for low Reynolds number airfoils," in Low Reynolds number aerodynamics, Springer, 1989, pp. 1–12.

[18] P. Castonguay and S. Nadarajah, "Effect of shape parameterization on aerodynamic shape optimization," in 45th AIAA Aerospace Sciences Meeting and Exhibit, 2007, p. 59.

[19] S. Nadarajah, P. Castonguay, and A. Mousavi, "Survey of shape parameterization techniques and its effect on three-dimensional aerodynamic shape optimization," in 18th AIAA computational fluid dynamics conference, 2007, p. 3837.

[20] V. Sripawadkul, M. Padulo, and M. Guenov, "A comparison of airfoil shape parameterization techniques for early design optimization," in 13th AIAA/ISSMO multidisciplinary analysis optimization conference, 2010, p. 9050.

[21] K. M. Selvan, "On the effect of shape parametrization on airfoil shape optimization," Int. J. Res. Eng. Technol., vol. 4, no. 2, pp. 123–133, 2015.

[22] D. Kiousis and Fons, "parsecairfoil.py," 2015. [Online]. Available: https://github.com/dqsis/parsec-airfoils. [Accessed: 10-Apr-2022].