JOURNAL OF MATHEMATICAL SCIENCES AND MODELLING

ISSN: 2636-8692

VOLUME VII
ISSUE II

VOLUME VII ISSUE II
ISSN 2636-8692

August 2024
http://dergipark.gov.tr/jmsm

JOURNAL OF MATHEMATICAL SCIENCES AND MODELLING

Language Editor

Technical Editor

Contents

Journal of Mathematical Sciences and Modelling, 7(2) (2024) 51-59
Research Article

Journal of Mathematical Sciences and Modelling

Journal Homepage: www.dergipark.gov.tr/jmsm
ISSN 2636-8692
DOI: https://doi.org/10.33187/jmsm.1472066

An SIR Model of Influenza with the Effects of Treatment and Vaccination

Elif Demir[1*] **and Canan Vural**[2]

[1]*Department of Mathematics, Faculty of Science and Arts, Yildiz Technical University, Istanbul, Türkiye*
[2]*Department of Mathematics, Faculty of Science, Istanbul University, Istanbul, Türkiye*
[*]*Corresponding author*

Article Info

Keywords: *Basic reproduction number, Fractional SIR model, Influenza, Rates of treatment and vaccination, Stability analysis*
2010 AMS: *26A33*
Received: *22 April 2024*
Accepted: *29 May 2024*
Available online: *11 June 2024*

Abstract

We produced an SIR model of influenza which is a global infectious disease, by using Caputo fractional derivative. In this model, we separated S and I into different groups. Separation is made according to the group of people in S who get vaccinated and are protected from influenza, also people in S who get vaccinated but are not protected besides people in S who do not get vaccinated. Furthermore, infected people are separated as treated and untreated people in I. We did stability analysis of the model and produced the basic reproduction number. We emphasized the importance of influenza vaccine and treatment for infected people by varying the values of the parameters and was shown with graphics.

1. Introduction

Influenza is the virus that represents a persistent and significant threat to global public health, responsible for deaths of nearly half a million people annually worldwide [1]. Annually, approximately 5% to 10% of adults and 20% to 30% of children are infected [2]. Individuals with chronic diseases of these patients are at an increased risk of developing greater morbidity or mortality when infected by the influenza virus due to poor health conditions and impaired immune systems [3]. Furthermore, influenza-related hospitalization and mortality rates were significantly higher in individuals with chronic diseases [4]. The most effective method of preventing influenza is vaccination, which is recommended for all individuals aged six months or older, including pregnant and postpartum women, unless there are contraindications [1]. Thomas Francis and Jonas Salk at the University of Michigan developed the first inactivated influenza vaccine and this vaccine is licensed in 1945 for wider use [5]. The influenza virus undergoes annual mutations, which necessitate the annual revision of the vaccine content. Vaccination should occur at the beginning of the influenza season, which typically begins in October [1]. The classic symptoms of influenza are fever, fatigue, cough and body aches [1]. In the outpatient setting, diagnosis can be made based on clinical presentation, with optional confirmatory diagnostic testing. This disease, which many patients can recover without any treatment but can be fatal for people in risk groups such as chronically ill patients, babies and the elderly people. Some people need to take antiviral treatment within 24 to 48 hours after the onset of the disease rather than drug-free treatment such as drinking plenty of water [6]. It is recommended that drugs be used in accordance with the instructions provided by the prescribing physician.
The model presented in this study incorporates parameters that elucidate the impact of vaccination and treatment on the epidemiological dynamics of the disease. Once the model has been generated, the stability conditions for the disease-free and endemic equilibrium points are determined separately. Moreover, the basic reproduction number is found specifically for this model. Different values were assigned to the model parameters in order to assess the impact of vaccination and treatment on the course of influenza.

Email addresses and ORCID numbers: edbilek@yildiz.edu.tr, 0000-0001-5973-9115 (E. Demir), canan.vural@istanbul.edu.tr, 0009-0000-6631-0085 (C. Vural),
Cite as "E. Demir, C. Vural, *An SIR model of influenza with the effects of treatment and vaccination*, J. Math. Sci. Model., 7(2) (2024), 51-59"

2. Model

We constructed a model as follows:

$$\begin{aligned} {}^{C}D_t^{\alpha} S &= \mu N - \frac{(1-ab)\beta}{N} SI - (\mu + ab)S \\ {}^{C}D_t^{\alpha} I &= \frac{(1-ab)\beta}{N} SI - \big(\mu + c\gamma + (1-c)\kappa\big)I \\ {}^{C}D_t^{\alpha} R &= abS + \big(c\gamma + (1-c)\kappa\big)I - \mu R \end{aligned}$$

with assumptions:

- The rates of birth and death are equal.
- $S(t) \geq 0$, $I(t) \geq 0$, $R(t) \geq 0$ at any time t .
- All parameters take values between 0 and 1. So all parameters are non-negative.
- $\gamma > \kappa$.

In Table 2.1 , we gave all parameters.

Parameter	Description
μ	Birth rate and death rate
β	Contact rate between susceptible people and infected people
a	Rate of vaccination in S
b	Rate of vaccine efficiency
c	Rate of treatment in I
γ	Recovery rate for treated people in I
κ	Recovery rate for untreated people in I

Table 2.1: Table of parameters

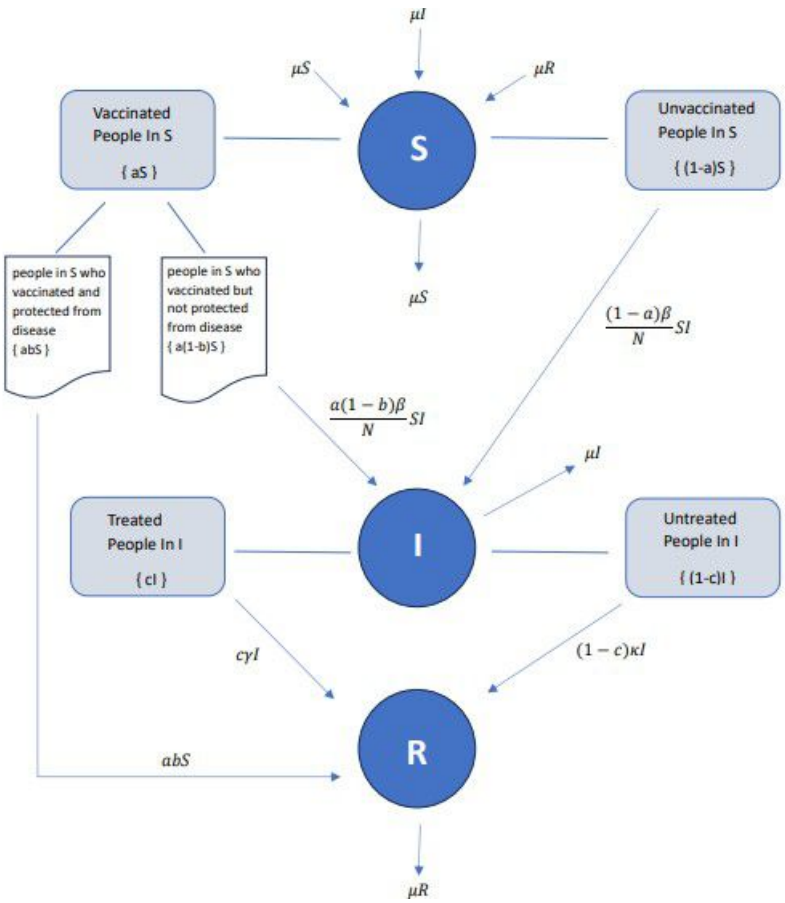


Figure 2.1: The model

3. Non-Negativity

Let us define the positive part of $\mathbb{R}^3$ as $\zeta \in \mathbb{R}^3 : \zeta \geq 0\}$ and $\zeta(t)$ as $(\zeta_1(t), \zeta_2(t), \zeta_3(t))^T$. We used Lemma 3.1 and Corollary 3.1 for proving the theorem that is related to non-negativity of the solutions [7].

Lemma 3.1 (Generalized Mean Value Theorem)**.** *Let $\psi(t)$ and $D^\alpha\psi(t)$ are continuous in the closed interval $[\rho,\sigma]$ for $0<\alpha\leq 1$. Then $\forall t\in[\rho,\sigma]$, $\psi(t)=\psi(\rho)+\frac{1}{\Gamma\alpha}D^\alpha\psi(\eta)(t-\rho)^\alpha$ holds with $0\leq\eta\leq t$ [8].*

Corollary 3.1. *Let $\psi(t)$ and $D^\alpha\psi(t)$ are continuous in the closed interval $[0,\sigma]$ for $0<\alpha\leq 1$. It can be seen from the lemma that $\psi(t)$ is non-decreasing $\forall t\in[0,\sigma]$ if $D^\alpha\psi(t)\geq 0$ for each $t\in(0,\sigma)$ and $\psi(t)$ is non-increasing $\forall t\in[0,\sigma]$ if $D^\alpha\psi(t)\leq 0$ for each $t\in(0,\sigma)$ [7].*

Theorem 3.1. *There exists a solution that unique and remains in the positive part of $\mathbb{R}^3$ for (1) such that $\zeta(t)=\big(S(t),I(t),R(t)\big)^T$.*

Proof. The existence and the uniqueness of $\zeta(t)$ in the interval $(0,\infty)$ can be obtained from [9] . It is necessary to explain that the domain $\mathbb{R}^3_+$ is positively invariant. Since
$D_t^\alpha S(t)=\mu N\geq 0$ where $S=0$,
$D_t^\alpha I(t)=0$ where $I=0$,
$D_t^\alpha R(t)=abS+\big(c\gamma+(1-c)\kappa\big)I\geq 0$ where $R=0$,
it can be obtained that by the corollary on each hyperplane bounding the non-negative octant, the vector field point into the positive part of $\mathbb{R}^3$. □

4. Stability Analysis

Two equilibrium points, the first one disease free equilibrium point and the second one endemic equilibrium point, are found as follows:

- $E_1=(\frac{\mu N}{\mu+ab},0,\frac{abN}{\mu+ab})$
- $E_2=\Big(\frac{(\mu+c\gamma+(1-c)\kappa)N}{(1-ab)\beta},N\big(\frac{\mu}{\mu+c\gamma+(1-c)\kappa}-\frac{(\mu+ab)}{(1-ab)\beta}\big),N\big(1+\frac{ab-c\gamma-(1-c)\kappa}{(1-ab)\beta}-\frac{\mu}{\mu+c\gamma+(1-c)\kappa}\big)\Big)$

by solving

$$\begin{aligned}\mu N-\frac{(1-ab)\beta}{N}\overline{SI}-(\mu+ab)\overline{S} &= 0\\ \frac{(1-ab)\beta}{N}\overline{SI}-\big(\mu+c\gamma+(1-c)\kappa\big)\overline{I} &= 0\\ ab\overline{S}+\big(c\gamma+(1-c)\kappa\big)\overline{I}-\mu\overline{R} &= 0.\end{aligned}$$

For this model the Jacobian matrix is obtained as follows:

$$\mathbf{J}=\begin{pmatrix}-\frac{(1-ab)\beta}{N}I-(\mu+ab) & -\frac{(1-ab)\beta}{N}S & 0\\ \frac{(1-ab)\beta}{N}I & \frac{(1-ab)\beta}{N}S-(\mu+c\gamma+(1-c)\kappa) & 0\\ ab & c\gamma+(1-c)\kappa & -\mu\end{pmatrix}.$$

Stability conditions for E_1: The Jacobian matrix is evaluated at disease free equilibrium point as follows :

$$\mathbf{J}|_{\mathbf{E_1}}=\begin{pmatrix}-(\mu+ab) & -\frac{\mu\beta(1-ab)}{\mu+ab} & 0\\ 0 & \frac{\mu\beta(1-ab)}{\mu+ab}-(\mu+c\gamma+(1-c)\kappa) & 0\\ ab & c\gamma+(1-c)\kappa & -\mu\end{pmatrix}.$$

Basic reproduction number :

Definition 4.1 (Next Generation Matrix Method, [10])**.** *Assume that $\mathscr{F}_i(x)$ is the new infections appearence rate in compartment i , $\mathscr{V}_i^+(x)$ is the transfer rate of individuals into compartment i by all other means and $\mathscr{V}_i^-(x)$ is the transfer rate of individuals out of compartment i. R_0 is defined as the spectral radius of FV^{-1} where*

$$F=\Big(\ \frac{\partial\mathscr{F}_i}{\partial x_j}(x_0)\ \Big)$$

and

$$V=\Big(\ \frac{\partial\mathscr{V}_i}{\partial x_j}(x_0)\ \Big).$$

The basic reproduction number (R_0) is found as $R_0=\frac{(1-ab)\mu\beta}{(\mu+ab)(\mu+c\gamma+(1-c)\kappa)}$ by using Next Generation Matrix method where

$$\begin{aligned}\mathscr{F}(S,I) &= \frac{(1-ab)\beta}{N}SI,\\ \mathscr{V}(I) &= -\big(\mu+c\gamma+(1-c)\kappa\big)I,\\ F &= \Big(\ \frac{(1-ab)\beta\mu}{\mu+ab}\ \Big),\\ V &= \big(\ -\big(\mu+c\gamma+(1-c)\kappa\big)\ \big).\end{aligned}$$

Stability analysis of the model can be done by using Matignon's conditions.

Theorem 4.2 (Matignon's Conditions)**.** *An equilibrium point is locally asymptotically stable if* $|arg(\lambda_i)| > \alpha\frac{\pi}{2}$ *for* $i = 1,2,3$ *where* λ_i*'s are the eigenvalues of Jacobian matrix that evaluated at the equilibrium point [11].*

Theorem 4.3. *If* $R_0 < 1$ *holds, then* E_1 *is locally asymptotically stable.*

Proof. Characteristic equation of $J|_{E_1}$ is obtained as follows:

$$\begin{aligned}
P(\lambda) &= \lambda^3 + \Big(\frac{3\mu^2+4\mu ab+a^2b^2-\mu\beta+\mu ab\beta+\mu c\gamma+(1-c)\kappa+abc\gamma+(1-c)\kappa}{\mu+ab}\Big)\lambda^2 \\
&+ \Big(\frac{-2\mu^2\beta+2\mu^2ab\beta+3\mu^3+5\mu^2ab+2\mu^2c\gamma+(1-c)\kappa+3\mu abc\gamma+(1-c)\kappa+2\mu a^2b^2-\mu ab\beta+\mu a^2b^2\beta+a^2b^2c\gamma+(1-c)\kappa}{\mu+ab}\Big)\lambda \\
&+ \Big(\frac{-\mu^3\beta+\mu^3ab\beta+\mu^4+\mu^3ab+\mu^3c\gamma+(1-c)\kappa+\mu^2abc\gamma+(1-c)\kappa-\mu^2ab\beta+\mu^2a^2b^2\beta+\mu^3ab}{\mu+ab} \\
&+ \frac{\mu^2a^2b^2+\mu^2abc\gamma+(1-c)\kappa+\mu a^2b^2c\gamma+(1-c)\kappa}{\mu+ab}\Big) = 0.
\end{aligned}$$

Eigenvalues of $J|_{E_1}$ are found by solving $P(\lambda) = 0$ as follows:

- $\lambda_1 = -\mu$
- $\lambda_2 = -(\mu+ab)$
- $\lambda_3 = \frac{-\mu ab+\mu\beta-\mu(c\gamma+(1-c)\kappa)-ab(c\gamma+(1-c)\kappa)-\mu^2-\mu ab\beta}{\mu+ab}$

Since all eigenvalues that obtained are real, according to the Matignon's Conditions the stability condition for E_1 is that all eigenvalues are negative. It is obvious that λ_1 and λ_2 are already negative.

$$\begin{aligned}
\frac{(1-ab)\mu\beta}{(\mu+ab)(\mu+c\gamma+(1-c)\kappa)} < 1 \quad &\Rightarrow \quad (1-ab)\mu\beta < (\mu+ab)(\mu+c\gamma+(1-c)\kappa) \\
&\Rightarrow \quad \mu\beta-\mu ab\beta < \mu^2+\mu c\gamma+(1-c)\kappa+\mu ab+abc\gamma+(1-c)\kappa \\
&\Rightarrow \quad -\mu ab+\mu\beta-\mu(c\gamma+(1-c)\kappa)-ab(c\gamma+(1-c)\kappa)-\mu^2-\mu ab\beta < 0
\end{aligned}$$

When $R_0 < 1$, since $-\mu ab+\mu\beta-\mu(c\gamma+(1-c)\kappa)-ab(c\gamma+(1-c)\kappa)-\mu^2-\mu ab\beta < 0$ and $0 < \mu+ab$, λ_3 is negative. □

Stability conditions for E_2 **:** The Jacobian matrix is evaluated at endemic point as follows :

$$\mathbf{J}|_{\mathbf{E_2}} = \begin{pmatrix} -\frac{\mu\beta(1-ab)}{\mu+c\gamma+(1-c)\kappa} & -(\mu+c\gamma+(1-c)\kappa) & 0 \\ \frac{\mu\beta(1-ab)}{\mu+c\gamma+(1-c)\kappa}-(\mu+ab) & 0 & 0 \\ ab & c\gamma+(1-c)\kappa & -\mu \end{pmatrix}.$$

Theorem: Characteristic equation of $J|_{E_2}$ is obtained as follows :

$$\begin{aligned}
P(\lambda) &= \lambda^3 + \Big(\frac{\mu^2+\mu(c\gamma+(1-c)\kappa)+\mu\beta-\mu ab\beta}{\mu+c\gamma+(1-c)\kappa}\Big)\lambda^2 \\
&+ \Big(\frac{2\mu^2\beta-2\mu^2ab\beta-\mu^2ab-\mu(c\gamma+(1-c)\kappa)^2-2\mu^2(c\gamma+(1-c)\kappa)-ab(c\gamma+(1-c)\kappa)^2-\mu^3-2\mu ab(c\gamma+(1-c)\kappa)}{\mu+c\gamma+(1-c)\kappa} \\
&+ \frac{\mu\beta(c\gamma+(1-c)\kappa)-\mu ab\beta(c\gamma+(1-c)\kappa)}{\mu+c\gamma+(1-c)\kappa}\Big)\lambda \\
&+ \Big(\frac{-\mu^3ab+\mu^3\beta-\mu^2(c\gamma+(1-c)\kappa)^2-2\mu^3(c\gamma+(1-c)\kappa)-\mu ab(c\gamma+(1-c)\kappa)^2-\mu^4}{\mu+c\gamma+(1-c)\kappa} \\
&+ \frac{-2\mu^2ab(c\gamma+(1-c)\kappa)+\mu^2\beta(c\gamma+(1-c)\kappa)-\mu^3ab\beta-\mu^2ab\beta(c\gamma+(1-c)\kappa)}{\mu+c\gamma+(1-c)\kappa}\Big) = 0.
\end{aligned}$$

Definition 4.4 ([7])**.** *The discriminant of the characteristic equation is defined as:*

$$D\big(P(\lambda)\big) = 18a_1a_2a_3+(a_1a_2)^2-4a_3a_1^3-4a_2^3-27a_3^2$$

where characteristic equation of the form

$$P(\lambda) = \lambda^3+a_1\lambda^2+a_2\lambda+a_3 = 0.$$

Discriminant of $P(\lambda)$ is evaluated as follows :

$$\begin{aligned} D\big(P(\lambda)\big) &= \frac{\big(\mu^2 ab+\mu(c\gamma+(1-c)\kappa)^2+\mu^2(c\gamma+(1-c)\kappa)+ab(c\gamma+(1-c)\kappa)^2+2\mu ab(c\gamma+(1-c)\kappa)}{(\mu+c\gamma+(1-c)\kappa)^4} \\ &- \frac{-\mu\beta(c\gamma+(1-c)\kappa)+\mu ab\beta(c\gamma+(1-c)\kappa)\big)^2\big(4\mu^4+4\mu^3 ab\beta+4\mu^3 ab-4\mu^3\beta+12\mu^3(c\gamma+(1-c)\kappa)}{(\mu+c\gamma+(1-c)\kappa)^4} \\ &- \frac{\mu^2a^2b^2\beta^2-2\mu^2ab\beta^2+8\mu^2ab\beta(c\gamma+(1-c)\kappa)+12\mu^2ab(c\gamma+(1-c)\kappa)+\mu^2\beta^2-8\mu^2\beta(c\gamma+(1-c)\kappa)+12\mu^2(c\gamma+(1-c)\kappa)^2}{(\mu+c\gamma+(1-c)\kappa)^4} \\ &- \frac{4\mu ab\beta(c\gamma+(1-c)\kappa)^2+12\mu ab(c\gamma+(1-c)\kappa)^2-4\mu\beta(c\gamma+(1-c)\kappa)^2+4\mu(c\gamma+(1-c)\kappa)^3+4ab(c\gamma+(1-c)\kappa)^3\big)}{(\mu+c\gamma+(1-c)\kappa)^4} \end{aligned}$$

Theorem 4.5. *Fractional Routh Hurwitz Conditions [12]:*

- *If* $D\big(P(\lambda)\big)>0$ *, then locally asymptotically stability conditions for the equilibrium point are* $a_1>0$ *,* $a_3>0$ *,* $a_1a_2>a_3$*.*
- *If* $D\big(P(\lambda)\big)<0$ *,* $a_1\geq 0$ *,* $a_2\geq 0$ *,* $a_3>0$ *then the equilibrium point is locally asymptotically stable for* $\alpha<\frac{2}{3}$*.*
- *If* $D\big(P(\lambda)\big)<0$ *,* $a_1<0$ *,* $a_2<0$ *and* $\alpha>\frac{2}{3}$ *then all roots of the characteristic equation of the endemic equilibrium point satisfy the condition* $|arg(\lambda_i)|<\alpha\frac{\pi}{2}$*.*

Taking into account the Fractional Routh Hurwitz Conditions, it can be seen that the conditions of stability are as follows:

- If $D\big(P(\lambda)\big)>0$, then locally asymptotically stability conditions for the equilibrium point are
  $\left(\frac{\mu^2+\mu(c\gamma+(1-c)\kappa)+\mu\beta-\mu ab\beta}{\mu+c\gamma+(1-c)\kappa}\right)>0,$
  $\left(\frac{-\mu^3ab+\mu^3\beta-\mu^2(c\gamma+(1-c)\kappa)^2-2\mu^3(c\gamma+(1-c)\kappa)-\mu ab(c\gamma+(1-c)\kappa)^2-\mu^4-2\mu^2ab(c\gamma+(1-c)\kappa)+\mu^2\beta(c\gamma+(1-c)\kappa)-\mu^3ab\beta-\mu^2ab\beta(c\gamma+(1-c)\kappa)}{\mu+c\gamma+(1-c)\kappa}\right)>0,$
  $\left(\frac{\mu^2+\mu(c\gamma+(1-c)\kappa)+\mu\beta-\mu ab\beta}{\mu+c\gamma+(1-c)\kappa}\right)$
  $\left(\frac{2\mu^2\beta-2\mu^2ab\beta-\mu^2ab-\mu(c\gamma+(1-c)\kappa)^2-2\mu^2(c\gamma+(1-c)\kappa)-ab(c\gamma+(1-c)\kappa)^2-\mu^3-2\mu ab(c\gamma+(1-c)\kappa)+\mu\beta(c\gamma+(1-c)\kappa)-\mu ab\beta(c\gamma+(1-c)\kappa)}{\mu+c\gamma+(1-c)\kappa}\right)>$
  $\left(\frac{-\mu^3ab+\mu^3\beta-\mu^2(c\gamma+(1-c)\kappa)^2-2\mu^3(c\gamma+(1-c)\kappa)-\mu ab(c\gamma+(1-c)\kappa)^2-\mu^4-2\mu^2ab(c\gamma+(1-c)\kappa)+\mu^2\beta(c\gamma+(1-c)\kappa)-\mu^3ab\beta-\mu^2ab\beta(c\gamma+(1-c)\kappa)}{\mu+c\gamma+(1-c)\kappa}\right).$
- If $D\big(P(\lambda)\big)<0$, $\left(\frac{\mu^2+\mu(c\gamma+(1-c)\kappa)+\mu\beta-\mu ab\beta}{\mu+c\gamma+(1-c)\kappa}\right)\geq 0$,
  $\left(\frac{2\mu^2\beta-2\mu^2ab\beta-\mu^2ab-\mu(c\gamma+(1-c)\kappa)^2-2\mu^2(c\gamma+(1-c)\kappa)-ab(c\gamma+(1-c)\kappa)^2-\mu^3-2\mu ab(c\gamma+(1-c)\kappa)+\mu\beta(c\gamma+(1-c)\kappa)-\mu ab\beta(c\gamma+(1-c)\kappa)}{\mu+c\gamma+(1-c)\kappa}\right)\geq 0,$
  $\left(\frac{-\mu^3ab+\mu^3\beta-\mu^2(c\gamma+(1-c)\kappa)^2-2\mu^3(c\gamma+(1-c)\kappa)-\mu ab(c\gamma+(1-c)\kappa)^2-\mu^4-2\mu^2ab(c\gamma+(1-c)\kappa)+\mu^2\beta(c\gamma+(1-c)\kappa)-\mu^3ab\beta-\mu^2ab\beta(c\gamma+(1-c)\kappa)}{\mu+c\gamma+(1-c)\kappa}\right)>0$
  then the equilibrium point is locally asymptotically stable for $\alpha<\frac{2}{3}$.
- If $D\big(P(\lambda)\big)<0$, $\left(\frac{\mu^2+\mu(c\gamma+(1-c)\kappa)+\mu\beta-\mu ab\beta}{\mu+c\gamma+(1-c)\kappa}\right)<0$,
  $\left(\frac{2\mu^2\beta-2\mu^2ab\beta-\mu^2ab-\mu(c\gamma+(1-c)\kappa)^2-2\mu^2(c\gamma+(1-c)\kappa)-ab(c\gamma+(1-c)\kappa)^2-\mu^3-2\mu ab(c\gamma+(1-c)\kappa)+\mu\beta(c\gamma+(1-c)\kappa)-\mu ab\beta(c\gamma+(1-c)\kappa)}{\mu+c\gamma+(1-c)\kappa}\right)<0$
  and $\alpha>\frac{2}{3}$ then all roots of the characteristic equation of the endemic equilibrium point satisfy the condition $|arg(\lambda_i)|<\alpha\frac{\pi}{2}$.

5. Numerical Approximation

We used Fractional Backward Euler Method to give numerical results with graphics.
Fractional backward Euler method. [13] The solution of

$$\begin{cases} D^\alpha f(t)=\Big(g\big(t,f(t)\big)\Big) \\ f(0)=f_0 \end{cases}$$

is as follows by using fractional backward Euler formula :

$$f_{k+1}=f_0+h^\alpha\sum_{j=0}^{k}a_{j,k+1}g(t_{j+1},f_{j+1}),\quad k=0,1,...,K-1.$$

Here,

$$a_{j,k+1}=\frac{(k-j+1)^\alpha-(k-j)^\alpha}{\Gamma(1+\alpha)},\quad k=0,1,...,K-1,\quad j=0,1,...,k.$$

The fractional backward Euler formula for the model is obtained as

$$\begin{aligned} S_{k+1} &= S_0+h^\alpha\sum_{j=0}^{k}a_{j,k+1}\big(\mu N-\frac{(1-ab)\beta}{N}S_{j+1}I_{j+1}-(\mu+ab)S_{j+1}\big) \\ I_{k+1} &= I_0+h^\alpha\sum_{j=0}^{k}a_{j,k+1}\big(\frac{(1-ab)\beta}{N}S_{j+1}I_{j+1}-\big(\mu+c\gamma+(1-c)\kappa\big)I_{j+1}\big) \\ R_{k+1} &= R_0+h^\alpha\sum_{j=0}^{k}a_{j,k+1}\big(abS_{j+1}+\big(c\gamma+(1-c)\kappa\big)I_{j+1}-\mu R_{j+1}\big) \end{aligned}$$

where

$$a_{j,k+1}=\frac{(k-j+1)^\alpha-(k-j)^\alpha}{\Gamma(1+\alpha)},\quad k=0,1,...,K-1,\quad j=0,1,...,k.$$

6. Case Studies and Numerical Results

This study demonstrates the impact of vaccines and treatments on influenza through graphs generated by adjusting model parameters.

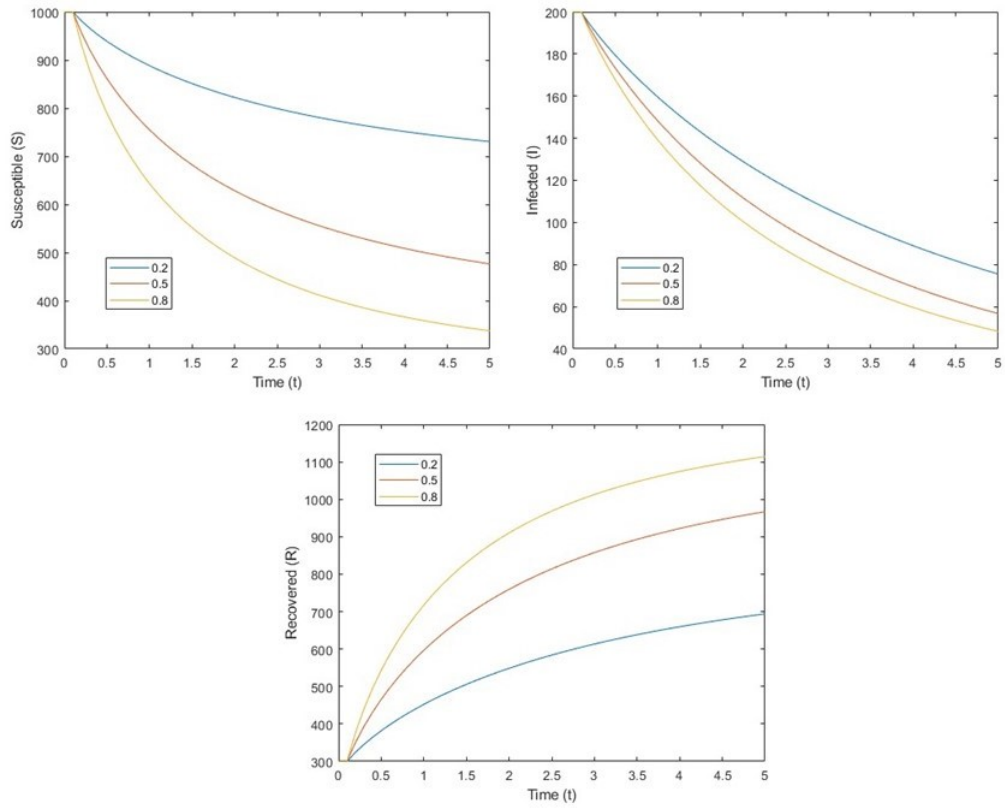


Figure 6.1: Graphs obtained by varying parameter a

In the graphs in Figure 6.1, the curves labelled "0.2" are for $a = 0.2, b = 0.7, c = 0.4, \mu = 0.1, \gamma = 0.6, \kappa = 0.3, \beta = 0.5$. $R_0 = 0.3445512821 < 1$ for this case. So disease free equilibrium point is locally asymptotically stable. But discriminant of $P(\lambda)$ for endemic equilibrium point is obtained as 0.0021 and $a_1 = 0.1827$, $a_3 = -0.0082$, $a_1 a_2 = -0.0134$. Since $a_3 < 0$, endemic equilibrium point is unstable according to Fractional Routh Hurwitz Conditions.
The curves labelled "0.5" are for $a = 0.5, b = 0.7, c = 0.4, \mu = 0.1, \gamma = 0.6, \kappa = 0.3, \beta = 0.5$. $R_0 = 0.1388888889 < 1$ for this case. So disease free equilibrium point is locally asymptotically stable. But discriminant of $P(\lambda)$ for endemic equilibrium point is obtained as 0.0317 and $a_1 = 0.1625$, $a_3 = -0.0201$, $a_1 a_2 = -0.0317$. Since $a_3 < 0$, endemic equilibrium point is unstable according to Fractional Routh Hurwitz Conditions.
The curves labelled "0.8" are for $a = 0.8, b = 0.7, c = 0.4, \mu = 0.1, \gamma = 0.6, \kappa = 0.3, \beta = 0.5$. $R_0 = 0.0641025641 < 1$ for this case. So disease free equilibrium point is locally asymptotically stable. But discriminant of $P(\lambda)$ for endemic equilibrium point is obtained as 0.1280 and $a_1 = 0.1423$, $a_3 = -0.0321$, $a_1 a_2 = -0.0451$. Since $a_3 < 0$, endemic equilibrium point is unstable according to Fractional Routh Hurwitz Conditions.
The graphs represent the case where parameter a values change and other parameters remain constant. Figure 6.1 demonstrate that as the value of the parameter a increases, the number of susceptible individuals is getting lower, the number of infected individuals is getting lower, and the number of recovered individuals is getting higher. This means that as individuals get vaccinated, transmission is reduced and therefore the number of infected is getting lower , according to our model . People in S who vaccinated and protected from the disease and do not get sick so they go to group R and the number of recovered individuals is getting higher in this case ; since people are significantly go to group R and protected from the disease, their possibilities of getting sick decrease, that is, the number of people in S is getting lower. In this comparison,it is shown that vaccination of susceptible people significantly reduces transmission.

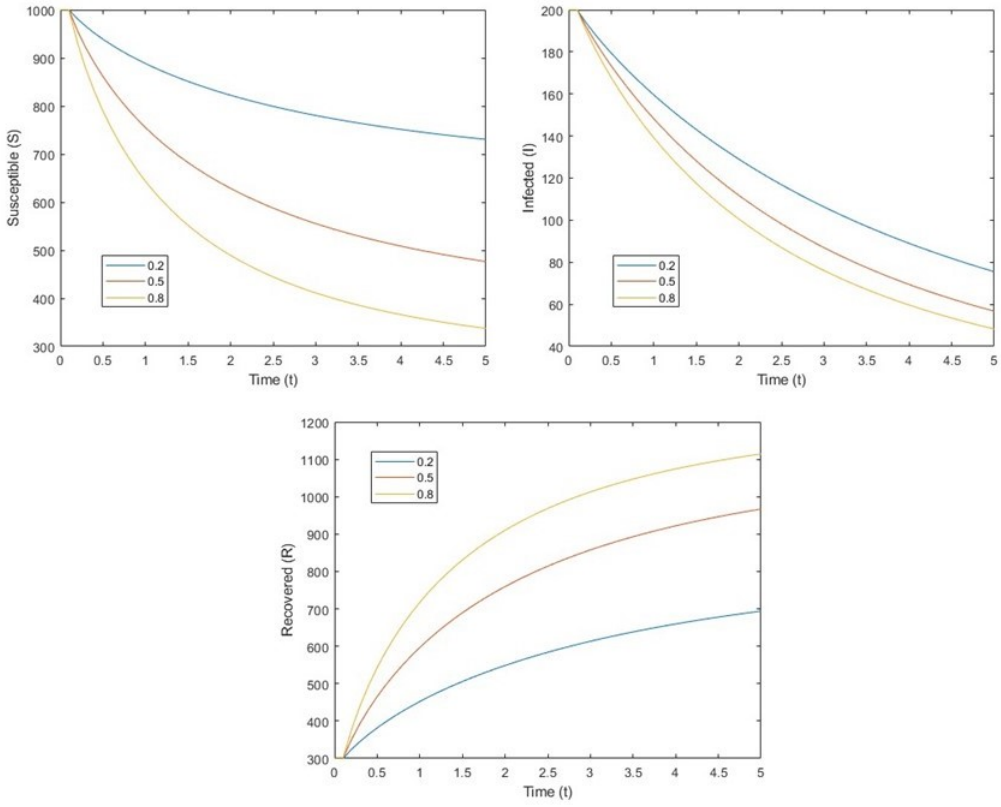


Figure 6.2: Graphs obtained by varying parameter b

In the graphs in Figure 6.2, the curves labelled "0.2" are for and $a = 0.7, b = 0.2, c = 0.4, \mu = 0.1, \gamma = 0.6, \kappa = 0.3, \beta = 0.5$. $R_0 = 0.3445512821 < 1$ for this case. So disease free equilibrium point is locally asymptotically stable. But discriminant of $P(\lambda)$ for endemic equilibrium point is obtained as 0.0021 and $a_1 = 0.1827$, $a_3 = -0.0082$, $a_1 a_2 = -0.0134$. Since $a_3 < 0$, endemic equilibrium point is unstable according to Fractional Routh Hurwitz Conditions.
The curves labelled $a = 0.7, b = 0.5, c = 0.4, \mu = 0.1, \gamma = 0.6, \kappa = 0.3, \beta = 0.5$. $R_0 = 0.1388888889 < 1$ for this case. So disease free equilibrium point is locally asymptotically stable. But discriminant of $P(\lambda)$ for endemic equilibrium point is obtained as 0.0317 and $a_1 = 0.1625$, $a_3 = -0.0201$, $a_1 a_2 = -0.0317$. Since $a_3 < 0$, endemic equilibrium point is unstable according to Fractional Routh Hurwitz Conditions.
The curves labelled "0.8" are for $a = 0.7, b = 0.8, c = 0.4, \mu = 0.1, \gamma = 0.6, \kappa = 0.3, \beta = 0.5$. $R_0 = 0.0641025641 < 1$ for this case. So disease free equilibrium point is locally asymptotically stable. But discriminant of $P(\lambda)$ for endemic equilibrium point is obtained as 0.1280 and $a_1 = 0.1423$, $a_3 = -0.0321$, $a_1 a_2 = -0.0451$. Since $a_3 < 0$, endemic equilibrium point is unstable according to Fractional Routh Hurwitz Conditions.
The graphs represent the case where parameter b values change and other parameters remain constant. Figure 6.2 demonstrate that as the value of the parameter b increases, the number of susceptible individuals is getting lower, the number of infected individuals is getting lower, and the number of recovered individuals is getting higher. This means that as individuals get more effective vaccines, transmission is reduced and therefore the number of infected is getting lower , according to our model. People in S who vaccinated and protected from the disease and do not get sick so they go to group R and the number of recovered individuals is getting higher in this case ; since people are significantly go to group R and protected from the disease, their possibilities of getting sick decrease, that is, the number of people in S is getting lower.

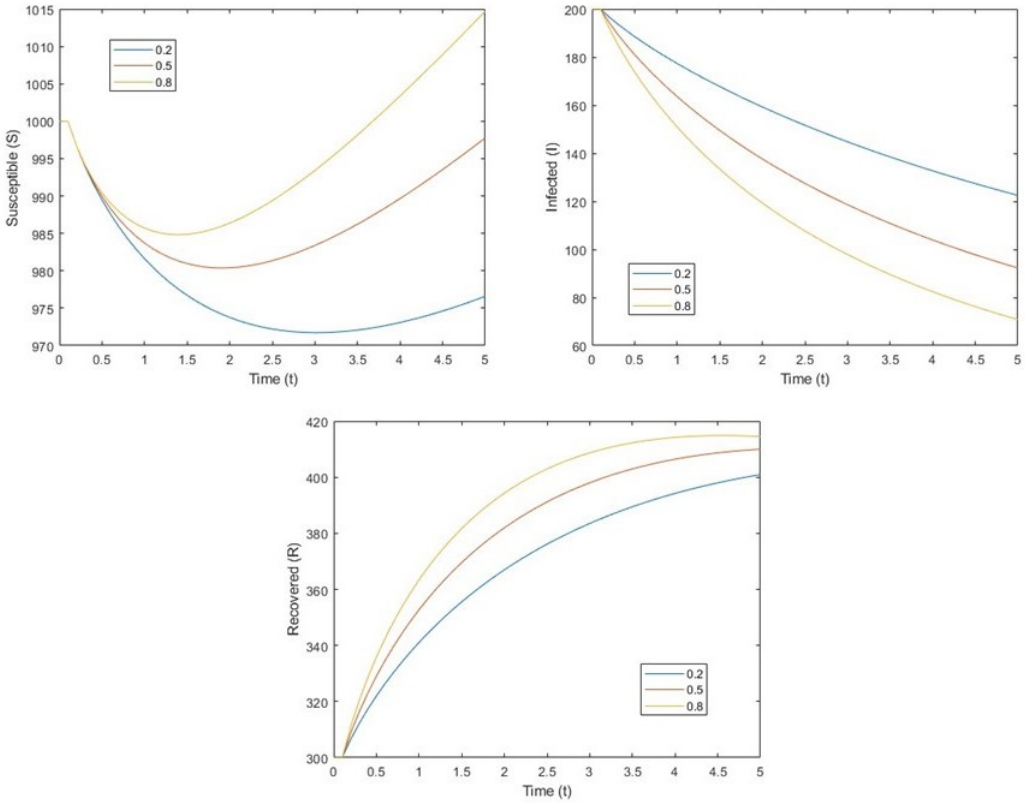


Figure 6.3: Graphs obtained by varying parameter c

In the graphs in Figure 6.3, the curves labelled "0.2" are for $a = 0.1, b = 0.1, c = 0.2, \mu = 0.1, \gamma = 0.6, \kappa = 0.3, \beta = 0.5$. $R_0 = 0.9782608696 < 1$ for this case. So disease free equilibrium point is locally asymptotically stable. But discriminant of $P(\lambda)$ for endemic equilibrium point is obtained as 0.00000006147321429 and $a_1 = 0.2076$, $a_3 = -0.00011$, $a_1 a_2 = 0.0020$. Since $a_3 < 0$, endemic equilibrium point is unstable according to Fractional Routh Hurwitz Conditions.
The curves labelled "0.5" are for $a = 0.1, b = 0.1, c = 0.5, \mu = 0.1, \gamma = 0.6, \kappa = 0.3, \beta = 0.5$. $R_0 = 0.8181818182 < 1$ for this case. So disease free equilibrium point is locally asymptotically stable. But discriminant of $P(\lambda)$ for endemic equilibrium point is obtained as 0.00000521 and $a_1 = 0.1900$, $a_3 = -0.0011$, $a_1 a_2 = -0.00038$. Since $a_3 < 0$, endemic equilibrium point is unstable according to Fractional Routh Hurwitz Conditions.
The curves labelled "0.8" are for $a = 0.1, b = 0.1, c = 0.8, \mu = 0.1, \gamma = 0.6, \kappa = 0.3, \beta = 0.5$. $R_0 = 0.703125 < 1$ for this case. So disease free equilibrium point is locally asymptotically stable. But discriminant of $P(\lambda)$ for endemic equilibrium point is obtained as 0.000031106 and $a_1 = 0.1773$, $a_3 = -0.0021$, $a_1 a_2 = -0.0023$. Since $a_3 < 0$, endemic equilibrium point is unstable according to Fractional Routh Hurwitz Conditions.
The graphs obtained represent the case where parameter c values change and other parameters remain constant. Figure 6.3 demonstrate that as the value of the parameters c increases, the number of susceptible individuals is getting higher, the number of infected individuals is getting lower, and the number of recovered individuals is getting higher. This means that as individuals in the infected group are treated they recovered and therefore the number of recovered people is getting higher ; since more infected individuals go to group R the number of infected people is getting lower.

7. Conclusion

In Figure 6.1, it is shown that getting vaccination of susceptible people significantly reduces transmission. In Figure 6.2, it is shown that getting efficient vaccination of susceptible people significantly reduces transmission.
In Figure 6.1 and Figure 6.2, parameter values except a and b are taken equal to emphasize that getting vaccine and getting effective vaccines are equally important. The correctness of this can be seen from the fact that exactly the same data are obtained in the graphs of Figure 6.1 and Figure 6.2.
In Figure 6.3, it is shown that treatment of infected people significantly increases recovery.

Article Information

Acknowledgements: The authors are grateful to The Scientific and Technological Research Council of Turkey (TUBITAK) for their financial support within the scope of "2210-A (2022/2) General Domestic Master Scholarship Programme" and Yildiz Technical University Scientific Research Projects Coordination Department with Project Number FYL-2023-5925.

Author's contributions: All authors contributed equally to the writing of this paper. All authors read and approved the final manuscript.

Conflict of interest disclosure: No potential conflict of interest was declared by the authors.

Supporting/Supporting organizations: No grants were received from any public, private or non-profit organizations for this research.

Ethical approval and participant consent: It is declared that during the preparation process of this study, scientific and ethical principles were followed and all the studies benefited from are stated in the bibliography.

Plagiarism statement: This article was scanned by the plagiarism program.

References

[1] C. Nypaver, C. Dehlinger, C. Carter, *Influenza and influenza vaccine: a review*, Journal of Midwifery & Women's Health, **66**(1) (2021), 45-53.
[2] A. D. Iuliano, et al., *Estimates of global seasonal influenza-associated respiratory mortality: A modelling study*, The Lancet, **391** (10127) (2018), 1285-1300.
[3] Y. Wang, et al., *Vaccination coverage with the pneumococcal and influenza vaccine among persons with chronic diseases in Shanghai, China, 2017*, BMC Public Health, **20** (2020), 1-9.
[4] R. Allard, et al, *Diabetes and the severity of pandemic influenza A (H1N1) infection*, Diabetes care, **33**(7) (2010), 1491-1493.
[5] https://www.who.int/news-room/spotlight/history-of-vaccination/history-of-influenza-vaccination?topicsurvey=ht7j2q)&gclid=Cj0KCQiAwbitBhDIARIsABfFYIJGDMPmzAm9bfYs7KULeumVIdTyBz8jYArZ40HX6oRQbYoQzhpXm1YaAqUqEALw_wcB
[6] https://grip.saglik.gov.tr/tr/tedavi.html
[7] R. Kumar, S. Kumar, *A new fractional modelling on susceptible-infected-recovered equations with constant vaccination rate*, Nonlinear Engineering, **3**(1) (2014), 11-19.
[8] Z. M. Odibat, N. T. Shawagfeh, *Generalized Taylor's formula*, Appl. Math. Comput., **186**(1) (2007), 286-293.
[9] W. Lin, *Global existence theory and chaos control of fractional differential equations*, J. Math. Anal. Appl., **332**(1) (2007), 709-726.
[10] P. Van den Driessche, J. Watmough, *Reproduction numbers and sub-threshold endemic equilibria for compartmental models of disease transmission*, Math. Biosci., **180**(1-2) (2002), 29-48.
[11] D. Matignon, *Stability results for fractional differential equations with applications to control processing*, Computational Engineering in Systems and Application, In: Multiconference, IMACS, IEEE-SMC, Lille, France, **2** (1996), 963-968.
[12] E. Ahmed, A.M.A. El-Sayed, H.A.A. El-Saka, *On some Routh–Hurwitz conditions for fractional order differential equations and their applications in Lorenz, Rössler Chua and Chen systems*, Phys. Lett. A, **358**(1) (2006), 1-4.
[13] P. Tomášek, *On Euler methods for Caputo fractional differential equations*, Arch. Math., **59** (2023), 287-294.

Journal of Mathematical Sciences and Modelling, 7(2) (2024) 60-74
Research Article

Journal of Mathematical Sciences and Modelling

Journal Homepage: www.dergipark.gov.tr/jmsm
ISSN: 2636-8692
DOI: https://doi.org/10.33187/jmsm.1484997

An Incident Management System Design to Protect Critical Infrastructures from Cyber Attacks

Uğur Gürtürk[1] **and Zeynep Gürkaş Aydın**[1*]

[1]*Department of Computer Engineering, Faculty of Engineering, Istanbul University-Cerrahpaşa, Istanbul, Türkiye*
[2]*Department of Computer Engineering, Faculty of Engineering, Istanbul University-Cerrahpaşa, Istanbul, Türkiye*
[*]*Corresponding author*

Article Info

Keywords: *Critical infrastructures, Cyber security, Information security, Log analysis*
2010 AMS: *68M25, 68M10, 68T01, 68T05*
Received: *16 May 2024*
Accepted: *4 June 2024*
Available online: *13 June 2024*

Abstract

In recent years, there has been a noticeable trend toward targeted threats to information security, where companies are now leveraging vulnerabilities and risks associated with widely used services in order to generate financial gain. Additionally, they implement numerous precautions and consistently carry out their tasks. One item that requires precautionary measures is the network devices utilized. Network devices in computer networks possess the capability to log events. These logs enable the identification of security events on the network and facilitate the implementation of precautionary measures. Various security measures can be implemented to handle such data. One of these measures is Security Information and Event Management (SIEM). It is a system that gathers and analyzes data from networks and security devices. SIEM is a technique employed to consolidate critical information within a cohesive structure. It allows for the correlation of events from different security devices, thereby improving the monitoring capabilities of cybersecurity operations centers. This study extensively covers the critical infrastructure-SIEM relationship, current studies, critical infrastructure, cyber security policies, and SIEM. Our system design was developed using the UNSW_NB15 dataset, a widely recognized dataset in cybersecurity due to its comprehensive and realistic representation of cyber threats. This dataset consists of data obtained from network traffic, various attack activities, and real-life modern normal scenarios, making it particularly relevant to our study. With the studies, a total of 10 different categories were analyzed, with the category consisting of nine types of attacks, namely Analysis, Backdoor, DoS, Exploits, Fuzzers, Generic, Reconnaissance, Shellcode, and Worms and Normal activities. The study is divided into two as the basic structure. The first step was carried out on Google Collaboratory, and then some experimental studies were carried out in Weka. Classifications were made using several methods, including Logistic Regression (LR), Extra Trees (XT), Support Vector Machines (SVM), Random Forest (RF), and Decision Trees (DT). These methods were chosen for their proven effectiveness in similar studies. In the application developed with Google Colabratory, we achieved 98.62% in Random Forest, 99.10% in Decision Trees, 98.87% in Logistic Regression, 95.13% success in Extra Trees and 99.12% success in Support Vector Machines. As a result of the studies and experiments carried out in Weka, we achieved 92.05% in Random Forest, 100% in Decision Trees, 100% in k-Nearest Neighbours, 100% in J48, 99.19% in Naive-Bayes and 99.35% in BayesNet achievements.

***Email addresses and ORCID numbers*: 23ugurgurturk23@gmail.com, 0000-0002-0695-0926 (U. Gürtürk), zeynepg@iuc.edu.tr, 0000-0002-4125-0589 (Z. Gürkaş Aydın)**

Cite as "U. Gürtürk, Z. Gürkaş Aydın, *An incident management system design to protect critical infrastructures from cyber attacks*, J. Math. Sci. Model., 7(2) (2024), 60-74"

1. Introduction

Security Information and Event Management (SIEM) refers to collecting and analyzing data from network and security devices, allowing for the correlation of events from various security devices. SIEM is a method that aggregates all essential information to enhance monitoring capabilities, commonly utilized by many Cyber Security Operation Centers today. It collects security data from network devices, servers, domain controllers, and more, storing and normalizing it to detect trends and threats and investigating user alerts by applying analytical methods. SIEM is a data collector, monitor, and reporting system that provides reporting capabilities to a security incident response team, matching specific rules to identify security issues and alerting them. Its primary focus is on ensuring information security, with its core objective being preserving the principles of Confidentiality, Integrity, and Availability. Missing to follow them may cause rejecting of the manuscript without further processing. Confidentiality protects information from unauthorized third parties due to its value, ranging from bank accounts, identity numbers, and credit card numbers to state secrets. Protecting this valuable data is the fundamental reason for information security, with encryption being the foremost solution when discussing confidentiality. Ensuring that data is accessible only to authorized personnel or third parties is crucial. Implementing file permissions and utilizing security protocols such as SSL/TLS (Secure Sockets Layer/Transport Layer Security) for communication over the internet are common solutions in today's world to restrict access to sensitive data and ensure confidentiality. Integrity, however, refers to protecting information from being altered. Ensuring data integrity involves maintaining data accuracy, which is crucial. Common methods to ensure integrity include hashing data using timestamps and comparing the hash value of the original message, among others. Availability refers to authorized individuals being able to access information when needed. It holds no value if the right people cannot access the information when necessary. Distributed Denial of Service (DDoS) attacks are prevalent attacks that target availability by bypassing the accessibility rule. SIEM software gathers log and event data produced by an organization's applications, security devices, and primary computer systems, consolidating them on a centralized platform. These software solutions also strive to identify interventions using the aforementioned fundamental principles. The data collected from antivirus events, firewall logs, and other sources is classified into categories such as malicious software activities and failed and successful login attempts. When a Security Information and Event Management (SIEM) system identifies a potential security threat while monitoring network activity, it produces an alert. It assigns a threat level according to predetermined rules. Figure 1.1. presents examples of security method evaluations.

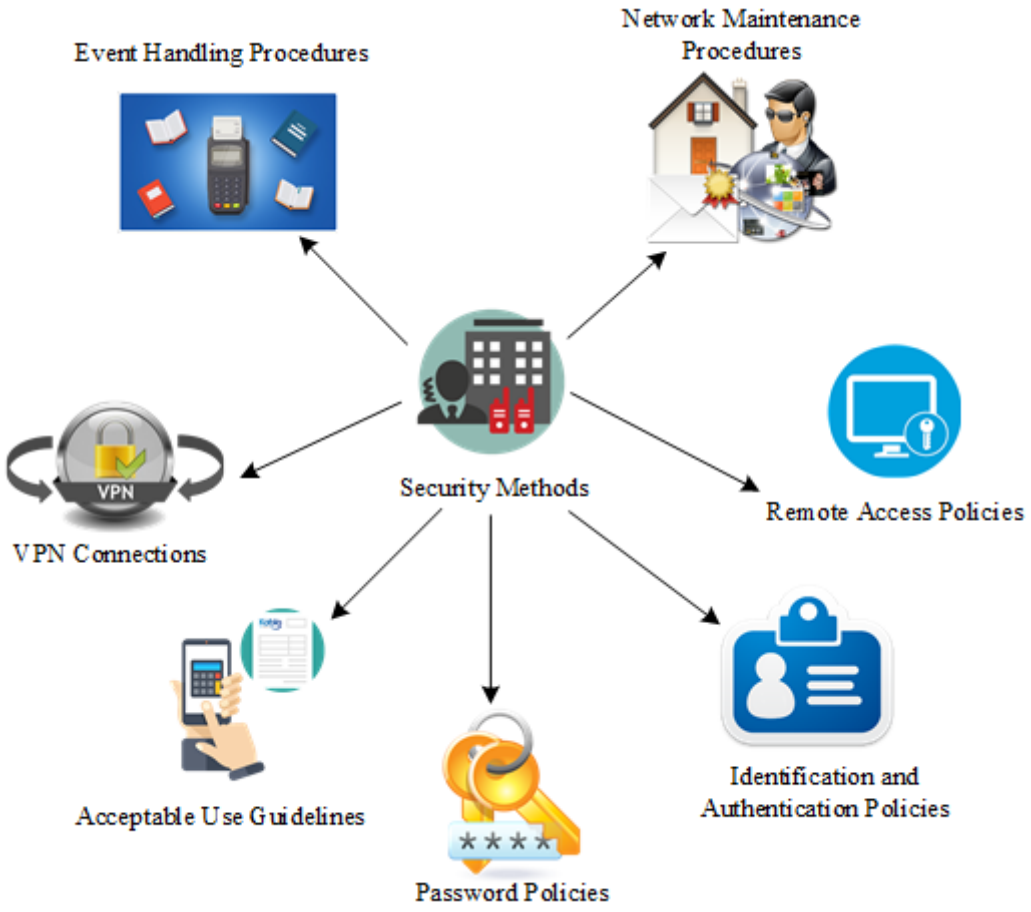


Figure 1.1: Security Method Evaluation

When examining SIEM tools, various types of vulnerabilities are noticeable. For instance, someone attempting to log in to an account 10 times in 10 minutes may not be seen as a problem, but an access request of 100 times in 10 minutes could be flagged as an attack attempt. SIEM detects such threats and generates security alerts. Its specialized dashboards and event management system enhance research efficiency and reduce time spent on false positives. When brought together and integrated, SIEM possesses a range of capabilities that provide comprehensive protection for organizations. Additionally, consolidating them into a single dashboard enables easier and more efficient management. The software used in this field allows security teams to gain insights into attacker tactics, techniques, procedures, and

known indicators of compromise through threat rules derived from information about security vulnerabilities. This can include User and Entity Behavior Analytics (UEBA), which monitors behaviors and activities to identify abnormal behaviors that could indicate a threat, lateral movement, and compromised security accounts. This is similar to the security analytics component that identifies anomalies in data to gather threat intelligence for threats that have not been encountered before. The managed rules component allows organizations to promptly respond to the most recent attacker techniques by utilizing analysts' nearly real-time updates. SIEM software generates alerts for an organization's security teams to promptly respond to threats, vulnerabilities, attacks, or suspicious behavior that it identifies. Certain software versions incorporate workflow and case management features to expedite investigations by automatically generating step-by-step investigation instructions in conjunction with searches and required actions. SIEM alerts can be customized to user requirements through customization in log management. Log management is a complex element of SIEM that encompasses three primary domains. Data collection involves consolidating an enormous amount of data from multiple applications and databases into a single location. SIEM collects event data from multiple sources throughout an organization's entire network. Data logs and flow data from users, applications, assets, cloud environments, and networks are gathered, stored, and examined instantly, enabling IT and security teams to centrally administer their network's event logs and network flow data automatically in a single central location. SIEM solutions can also incorporate third-party threat intelligence feeds to match internal security data with established threat signatures and profiles. Integrating with real-time threat feeds allows teams to identify or prevent new attack signatures. Data normalization: SIEM enables the comparison, correlation, and analysis of all different data types. It allows for classifying abnormal behaviors detected in the network across all connected users, devices, and applications and monitoring security incidents. Data analysis/security event correlation: This refers to identifying possible indicators of a data breach, threat, attack, or security vulnerability. Event correlation is a critical component of any SIEM solution. Using advanced analytics to identify and understand complex data models, event correlation provides insights to detect and mitigate potential threats to business security rapidly. Managers can be instantly alerted using customizable predefined correlation rules and take appropriate measures to mitigate them before they escalate into more significant security issues. SIEM solutions greatly enhance the Mean Time to Detect (MTTD) and Mean Time to Respond (MTTR) for IT security teams by eliminating the need for manual processes to analyze security events thoroughly. With its automated data collection and analysis capabilities, SIEM is a crucial tool for organizations that must gather and verify compliance data across their entire business infrastructure, especially those subject to different compliance formats. Given the fast pace at which the cyber security environment changes, organizations must depend on solutions that can identify and react to familiar and unfamiliar security risks. Organizations commonly select SIEM products that must adhere to various compliance formats because of their ability to collect and analyze data automatically. This enables them to gather and verify compliance data throughout the entire business infrastructure. Due to the swift rate of change in the cyber security landscape, organizations must depend on solutions capable of identifying and addressing familiar and unfamiliar security risks. Currently, organizations must choose between utilizing SIEM products or creating their own software [1–7].

2. Related Work

Computer networks are increasingly vulnerable to various types of cyber attacks due to security vulnerabilities. Therefore, cybersecurity aims to make networks as secure as possible by providing defense systems to detect suspicious activity. However, despite constantly updating their databases to detect cyber threats, traditional security systems (STS), such as firewalls, cannot fully ensure security. Therefore, the new guidelines for STS aim to utilize Machine Learning (ML) models to design reliable systems with higher detection rates and lower false alarm rates. Based on this approach, Moualla et al. (2021) proposed an approach to improve the performance of ML-based network intrusion detection systems using the UNSW-NB15 dataset. Considering studies using the UNSW-NB15 dataset and its importance in network security, they defined a new network STS. The proposed system is a dynamically scalable multi-class ML-based network STS. In the study, the Synthetic Minority Over-sampling Technique (SMOTE) was used to address the class imbalance problem in the dataset. Then, the Extremely Randomized Trees Classifier with the Gini Impurity criterion was used to select important features for each class in the dataset. An Extreme Learning Machines (ELM) model pre-trained as a binary classifier for each attack was used to detect attacks separately using the "One-versus-All" method. They demonstrated that the proposed system showed much better performance in terms of accuracy, false alarm rate, Receiver Operating Characteristic (ROC) curve, and Precision-Recall Curves (PRC) [8,9].

In the study by Zoghi et al. (2021), they addressed the class imbalance and class overlap problems that must be considered before using the UNSW-NB15 dataset. As part of the preparation for visual analysis, they applied preprocessing steps such as removing irrelevant features, normalizing features, and scaling features. After data preprocessing, they visualized the dataset using various visualization techniques to reveal and demonstrate class imbalance and class overlap issues, projecting the dataset onto 2D and 3D views using Principal Component Analysis (PCA). They then analyzed it using distribution plots, t-SNE, and K-means cluster distance maps. They argued that applying effective approaches to reduce the negative effects of any statistical or ML model on classification performance based on data is essential [10].

Aleesa et al. (2021) aimed to test the models of the UNSW-NB15 dataset by merging the entire dataset into a single file instead of separately testing them for each file, allowing models to be tested once. They then used attack types in the dataset as a new class, aiming to develop a multi-classified labeled dataset. With the dataset they developed, they investigated the performance of deep learning in both Binary and Multi-Class categories. They found that the proposed deep learning models achieved 99.59% accuracy in multi-class classification and 99.26% in binary classification [11]. Kocher et al. (2021) employed the UNSW-NB15 dataset to train classifiers, including KNN, Stochastic Gradient Descent (SGD), RF, LR, and NB, for the purpose of classification. A comparative analysis was conducted on the classifier performance, accuracy, Mean Squared Error (MSE), precision, recall, F1-Score, True Positive Rate (TPR), False Positive Rate (FPR), and feature selection technique. The UNSW-NB15 dataset was subjected to the Chi-Square filter-based feature selection technique to eliminate irrelevant and unnecessary features [12].

Mahalakshmi et al. (2021) demonstrated that the binary-encoded dataset showed maximum performance when using the Convolutional Neural Networks (CNN) deep learning method on the UNSW-NB15 dataset [13]. Iqbal et al. (2021) aimed for a more efficient machine learning approach to detect botnets in IoT networks using the PyCaret machine learning library and analyzing its overall performance [14]. Sharma et al. (2021) evaluated whether attack detection with machine learning methods, applying certain operations on the dataset using RF, XT, AdaBoost, and XGBoost methods, was accurate [15]. In the study by Sarhan et al. (2022), they transformed the UNSW-NB15, BoT-IoT, ToN-IoT, and CSE-CICIDS2018 datasets into new variants with the proposed NetFlow-based feature sets based on the XT classifier. They then compared the classification performance of NetFlow-based feature sets with registered feature sets provided with the original data

sets [16].

Pacheco et al. (2021) evaluated the security vulnerability against popular deep learning attack methods such as Multi-Layer Perceptron (MLP), DT, RF, and SVM on UNSW-NB15 and Bot-IoT datasets. They assessed the reliability of various machine learning classifiers [17]. Kılınçer et al. (2021) conducted a study in which they extensively reviewed literature studies that utilized commonly used datasets such as CSE-CIC IDS-2018, UNSW-NB15, ISCX-2012, NSL-KDD, and CIDDS-001 for the development of Security Testing Suites (STS). Additionally, they performed max-min normalization on these datasets and classified them using classical machine learning approaches such as SVM, KNN, and DT algorithms [18]. Kushwah et al. (2021) introduced a DDoS attack detection system based on Self-Adaptive Evolutionary Extreme Learning Machine (SaE-ELM) [19]. Roy et al. (2021) proposed a technique using SVM and NB algorithms, claiming to solve the classification problem of attack detection systems [20].

Ahsan et al. (2021) utilized CNN, LSTM, Bidirectional LSTM (Bi-LSTM), Gated Recurrent Units (GRU), and RF methods. They extracted accuracy rates and then reduced features using their proposed algorithm by filtering out insignificant variables, thus providing more accurate predictions [21]. Pooja et al. (2021) implemented an identity model based on deep learning methods and Bi-LSTM. The designed system was tested using KDDCUP-99 and UNSW-NB15 datasets, with the Bi-LSTM model yielding results with 99% accuracy for both datasets. Additionally, experiments were repeated by altering activation functions used in the network [22].

Thirimanne et al. (2021) aimed to identify the best machine learning algorithm for intrusion detection trained on NSL-KDD and UNSW-NB15 datasets and conducted a comparative analysis among six machine learning algorithms classified as supervised, semi-supervised, and unsupervised learning. The study revealed that supervised and semi-supervised machine learning algorithms outperformed unsupervised machine learning algorithms for both datasets. SVM and DNN perform better for NSL-KDD and UNSW-NB15, respectively [23]. Rani et al. (2022) proposed a deep neural network for addressing class imbalance. The network data underwent preprocessing through data transformation followed by min-max normalization using the Artificial Neural Networks (NN) method for analysis [24].

Okay et al. (2021) propose a methodology to reduce deficiencies in existing STSs for WLANs, aiming to create a more effective system that can dynamically detect unknown and complex attack variants. Two main contributions of the proposed methodology are claimed. The first contribution is using the Feature Selection Approach (FSAP) to reduce the number of features used, thus increasing the speed of attack detection. The second contribution is a hybrid attack detection approach using Signature and Anomaly-Based Attack Detection Technique (SABADT), which can detect attacks quickly and accurately. The proposed methodology was applied to KDD'99 and UNSW-NB15 datasets. The results were compared with existing machine learning techniques. The detection model was created using KDD'99 and UNSW-NB15 training datasets and tested on KDD'99 and UNSW-NB15 test datasets. The obtained accuracy rates of 99.65% and 99.17% were considered quite high compared to leading methods in the literature. Additionally, common tools were used to obtain a mixture of normal activities and existing attack behaviors to test new attacks. Different types of attacks were captured using Wireshark, and some of these captured attacks were used only in the testing phase. In this test scenario, attacks were detected with a 99.69% accuracy rate [25].

Sekhar et al. (2021) proposed a new Attack Detection Technique using Fruitfy Optimization with Deep Autoencoder. Firstly, missing values in the dataset were replaced using the Fuzzy C-Means Rough Parameter (FCMRP) algorithm, which handles uncertainty in datasets by leveraging fuzzy and rough clusters while preserving important information. Next, robust features were extracted from Autoencoder with multiple hidden layers. Finally, the obtained features were fed into a Backpropagation Neural Network (BPN) for classifying attacks. Additionally, neurons in the hidden layers of Deep Autoencoder were optimized using the population-based Fruitfy Optimization algorithm. Experiments were conducted on NSL KDD and UNSW NB15 datasets. The computational results of the proposed attack detection system using a BPN-based deep autoencoder were compared with NB, SVM, Radial Basis Function Networks (RBFN), BPN, and Autoencoder with Softmax [26].

With the rapid development of the Internet, cyber-attack methods have become more complex, causing increasingly significant damage worldwide. Therefore, in Yang's (2021) study, given the growing importance of timely detection of malicious behavior on the Internet as a significant security issue, a deep learning-based STS is proposed, implementing bidirectional LSTM architecture and using the UNSW-NB15 dataset for training and testing. Experimental tests in the study showed that the STS effectively detects known or unknown malicious behaviors in the current network environment [27].

Han et al. (2022) introduced an Intrusion Detection Hyperparameter Control System (IDHCS) that manages and trains a k-means clustering module as a reinforcement learning model using a DNN feature extractor and Proximal Policy Optimization (PPO). The IDHCS system utilizes a DNN feature extractor to extract the most significant features in the network environment. It then identifies unauthorized entries by employing k-means clustering. The reinforcement learning model, which combines PPO and iterative learning, is designed to automatically enhance performance in the network environment of IDHCS. System performance was assessed through experiments using the CICIDS2017 and UNSW-NB15 datasets. Achieving an F1 score of 0.96552 in CICIDS2017 and a score of 0.94268 in UNSW-NB15 demonstrates the level of performance attained. A series of experiments were conducted by merging the two datasets to establish a more comprehensive and intricate testing environment. The diversity of attack types in the experiment increased due to merging the datasets. Achieving an F1 score of 0.93567 in the combined dataset demonstrates a significant performance improvement of 97% to 99% compared to the CICIDS2017 and UNSW-NB15 datasets. The results indicate that the proposed Intrusion Detection and Host-based Control System (IDHCS) automates learning to detect new types of attacks and enhances the performance of Security Threat Scenarios (STS). This is achieved by utilizing unauthorized entry detection features independent of network environment changes and through continuous learning [28].

Al-Gethami et al. (2021) aimed to demonstrate that ML-based STSs' classification accuracy can be influenced by certain factors using DT, RF, SVM, ANN, and NB algorithms. The factors considered in the study were the method of using the dataset for training and testing, removal of outliers and extreme value instances, addition of mislabeled instances, and the use of ensemble learning techniques. The study showed various effects of these factors on classification accuracy; in some cases, the impact of noise in the data on the accuracy of the RF algorithm was demonstrated to have negative effects on classification accuracy. However, the negative effects of these factors were shown to result in significant improvement when applied to the UNSW-NB15 dataset with classification methods such as DT, RF, SVM, ANN, and NB [29].

Meliboev et al. (2022) proposed a DL method to implement an effective and adaptive STS using CNN, LSTM, Recurrent Neural Network (RNN), and Gated Recurrent Units (GRU) methods [30]. El-Sayed et al. (2021) proposed a suggestion for attack detection in intelligent transportation systems using only 20 features from the dataset. SVM, ANN, and NB algorithms were used as methods, and it was found that the proposed method increased accuracy [31]. Kim et al. (2021) introduced a model that utilizes LSTM, a machine learning technique, for

predicting the timing of attacks in the prediction domain of four different types of STS attacks. The UNSW-NB15 dataset was employed for this purpose. During training, the LSTM inputs for each attack comprised 80% and 90% of the examples. The epoch values were incremented from 1 to 26. The model accurately predicted the number of attack points for Back Door, DoS, Exploit, and Generic attacks. The output values produced by the model were examined to verify the precise timing of the attack. During the analysis of Dos, Backdoor, and Generic attacks, it was discovered that the expected attack occurrence was T[i]+2 time slots, in contrast to the actual attack timing of T[i]. Nevertheless, in the context of an Exploit attack, researchers observed a series of overlapping attacks and encountered challenges in accurately interpreting the numerical values generated by the LSTM model. Consequently, it became arduous to ascertain whether an attack had occurred. [32].

Hossain et al. (2021) evaluated the performance of four popular classifiers, DT, SVM, RF, and NB, using the Pandas and SKlearn libraries with Python language on the UNSW-NB15 dataset. They used the UNSW-NB15 dataset with 43 features comprehensively. The experimental results showed improved RF, DT, and NB accuracy compared to previously reported results by Apache Spark and its MLlib [33]. In Dutt et al. (2021), the authors extensively analyzed and examined two commonly used datasets in intrusion detection, namely KDD'99 and UNSW-NB15 datasets. Data preprocessing was performed for both datasets, considering missing, redundant, and noisy data, using the Weka data mining tool. A new dataset using a week's worth of network traffic was also introduced. This dataset was prepared considering resource consumption-based features, as these features played a significant role in detecting unauthorized entries. It was also emphasized that preprocessing is crucial in any predictive scenario applicable to intrusion detection systems. Data preprocessing for KDD'99 and UNSW-NB15 datasets was demonstrated using the Weka data mining tool [34].

Kim et al. (2021) introduced a deep learning model that utilizes LSTM and GRU on the UNSW-NB15 dataset to forecast the occurrence of attacks in a dataset for intrusion detection systems. The application of finite state machines was utilized to convert floating-point values into corresponding binary values, enhancing the model's accuracy. Consequently, they discovered that the precision of GRU and LSTM, measured by weighted F1 scores, was roughly 13% and 18% greater, respectively [35]. Intrusion detection systems (IDSs) are critical for protecting ICT infrastructures (STSs). Deep learning and machine learning are widely used to process high-dimensional, complex data to provide robust solutions for new attack types and complexity control. IDSs using unsupervised machine learning techniques detect and capture attack types such as known, unknown, and zero-day attacks. Sing et al. (2021) designed a structure using the concept of One-Class SVM (OCSVM) and active learning to detect threats without prior knowledge loss. The performance of this structure was tested using the CIC-IDS2017 dataset and compared with the UNSW-NB15 and KDD cup 99 datasets, demonstrating superior performance [36].

Silva et al. (2022) highlighted the lack of accurate evaluation, comparison, and distribution due to the scarcity of well-structured datasets in machine learning mechanisms to detect unauthorized entries on the network. They proposed a statistical analysis of the features in the four most commonly used datasets. It was concluded that the analyzed datasets should not be used as a comparison to create new anomaly-based mechanisms for intrusion detection systems. Instead, the correlation between features was analyzed to control outliers or data imbalance. DT, LR, RF, and XGBoost methods were applied in the analysis phase [37]. In the study by Priya et al. (2021), performance analysis and feature analysis of STS datasets were presented by comparing the Sigmoid classifier with datasets such as KDDCUP99, DARPA 1999, TWENTE 2008, UNIBS 2009, ISCX2012, NGIDS-DS 2016, UNSWNB15, and CICIDS2017 [38].

The study conducted by Man et al. (2021) introduced a network intrusion detection model based on machine learning. Initially, the study performed preprocessing on the dataset and then transformed it into images. More complex convolutional neural networks incorporating residual blocks were developed to capture important features effectively. Additionally, focal loss was employed to handle the class imbalance problem in the training set and improve the detection of minor attacks in the test set. The model employed batch normalization and global average pooling techniques to mitigate and enhance the accuracy of false detections. The experiments demonstrated that the suggested model has the potential to enhance the accuracy of attack detection in comparison to current models [39]. Ashiku et al. (2021) propose using deep learning architectures to develop an adaptable and flexible network STS for detecting and classifying network attacks. The study addresses network intrusion detection systems using a network traffic dataset that includes common cybersecurity vulnerabilities. The combined system merged with a semi-dynamic hyperparameter adjustment approach, showed significant improvements in multi-class models compared to similar deep learning-based network STS. The proposed approach achieved an overall accuracy of 95.4% and 95.6% for pre-segmented and user-defined multi-class classification, respectively [40].

Hooshmand et al. (2022) propose a model using one-dimensional CNN architecture. The approach initially separates network traffic data into Transmission Control Protocol (TCP), User Datagram Protocol (UDP), and other protocol categories. Each category is then addressed independently. Feature selection using the Chi-square technique was performed before training the model, followed by oversampling using the synthetic minority oversampling technique (SMOTE) to address the class imbalance. The proposed method achieved weighted average F-scores of 0.85, 0.97, 0.86, and 0.78 for TCP, UDP, OTHER, and ALL categories, respectively [41]. The study by Kamarudin et al. (2021) aims to improve the accuracy of detecting DoS attacks using ML by identifying the most important features in the dataset and analyzing abnormal network activities more accurately. Experimental results show the accuracy of the proposed system in detecting DoS attacks using the RF method, with accuracy rates of over 98.8% compared to traditional approaches for each dataset [42].

Magan-Carrión et al. (2021) presented a new methodology for ML-based Network STS called Reliable-NIDS (R-NIDS) and proposed a new dataset named UNK21. They collected the system from the three well-known attack datasets (UGR'16, USNW-NB15, and NLS-KDD), each from their own network environment, with different features and classes, using an existing data collection approach. They showed increased attack detection accuracy with this dataset [43]. The study by Sharma et al. (2021) aimed to increase attack detection accuracy on the UNSW-NB15 dataset using RF, XT, AdaBoost, and XGBoost methods. Comparative analysis of all classifiers used standard evaluation parameters, and graphs were utilized to examine outlier values in each feature. The obtained graphs provided insights into each feature's minimum, maximum, and median values for each class label. Graphical analysis of the data presented frequency distributions for each feature and explained the correlation between features. Dot plots revealed that the mean value of each feature was around zero for attack labels and varied significantly for normal class labels. The Random Forest classifier achieved the highest accuracy of 86.9%, while AdaBoost had the lowest accuracy despite minimal differences in performance among all classifiers, yielding nearly similar results [44].

In the study by Chew et al. (2021), ten important machine classifiers (ZeroR, Random Tree, REPtree, Decision Stump Adaboost, Bayesnet, NB, RF, SMO, and J48) were used to evaluate three selected NIDS datasets. The analysis revealed that the UNSWNB15 dataset is more suitable for attack detection in low-footprint scenarios, the CIDDS-001 dataset is suitable for detecting reconnaissance techniques, and GureKDDCup contains most features similar to the previous comparison KDDCup'99 and can be used as an alternative dataset [45]. Acharya

et al. (2021) discussed the effectiveness of various machine learning algorithms, including RF, J48, NB, BayesNet, Bagging, AdaBoost, and SVM, using Weka on network log traffic datasets from KDD99, UNSW-NB15, and CIC-IDS2017. They examined the impact of changing the output class counts of network attack datasets on sensitivity, TPR, FPR, Area Under Curve (AUC) of the ROC Curve, and misclassified percentage. The study showed that reducing the target class count increased the performance of machine learning classifiers and adding a highly correlated feature to the output class improved classifier performance [46]. In the study by Dlamini et al. (2021), they introduced a data generator model named DGM to improve anomaly detection accuracy in the anomaly detection domain. They conducted experiments on NSL-KDD and UNSW-NB15 datasets to demonstrate the effectiveness of their approach. They compared their method with the existing statistical approach SMOTE and found that DGM performed better [47].
Pavlov et al. (2021) addressed the requirements for creating datasets for use in unauthorized entry detection systems on networks and analyzed modern datasets. They created a requirement list for datasets used in test methods for identifying attacker groups, determined weights for requirements, and established a usability rating for modern datasets. They also proposed an alternative data source to meet requirements inadequately addressed by current datasets [48]. The study by Güler et al. (2021) utilized supervised learning classification-based algorithms, namely RNN, LSTM, and GRU, to compare their effectiveness in detecting network attacks using the UNSW-NB15 dataset. The study's primary objective was to evaluate the efficacy of deep learning algorithms and identify the most optimal model for detecting and classifying attacks. The models achieved accuracy values of 98%. The false positive rate (FPR) values for the RNN, LSTM, and GRU models were determined to be 0.014, 0.011, and 0.011, respectively [49].

3. Dataset and Methods

Data analysis can be defined as the process of transforming raw data into useful information. It leverages big databases to extract insights with predictive analytics, enabling companies, organizations, or individuals to focus on the most critical information in data warehouses. Data analysis possesses significant potential as an emerging technology that can forecast future trends and behaviors, thereby enabling businesses to make well-informed decisions. Data analysis employs advanced mathematical algorithms to segment data, evaluate the likelihood of future events, and facilitate effective decision-making and action. Existing software and hardware platforms can quickly incorporate data analysis techniques to improve the usefulness of current information sources and seamlessly integrate with new products and systems. The primary characteristics of data analysis are as follows:

- Automated discovery of patterns
- Prediction of potential outcomes
- Generation of practical insights
- Emphasis on big datasets and databases
- Proficiency in addressing complex inquiries that cannot be resolved using basic query and reporting methods.

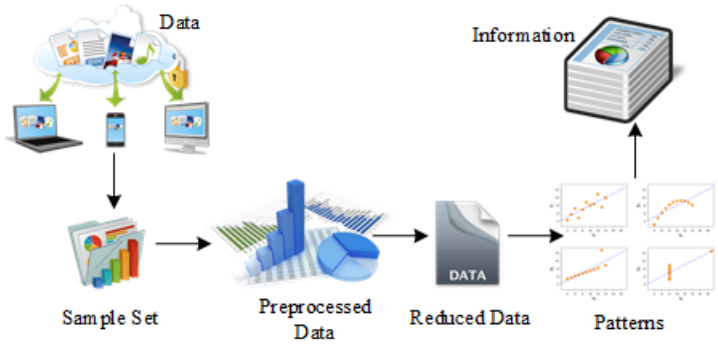


Figure 3.1: Steps of Data Analysis Process [50]

In Figure 3.1, the steps of the data analysis process are provided. Among the given steps, the first and most crucial step of data analysis is selecting data to be analyzed and making decisions to be derived from it. It is one of the most time-consuming steps in the data analysis process steps. In this step, the data generated in the system should be well-selected, and the analysis should be carried out meticulously to ensure the accuracy of the decision. Another important step for successful data analysis implementation is preprocessing, where data is prepared for later use. The success achieved at this stage significantly influences the success of the outcome. The proper and efficient execution of the preprocessing stage will lead to clear and definitive results.
In the data reduction phase, the sample dataset obtained from collected data undergoes a specific preprocessing stage to obtain useful and real information. In this phase, although the data has gone through a certain preprocessing, the data reduction process that will not be used in subsequent steps is carried out to bring it into the necessary format for later use. For the complete implementation of data analysis and the method, data mining methods are applied to the reduced data according to the purpose of the study. One or more known data analysis techniques can be applied to the reduced data at this stage. Moreover, different data analysis methods can be combined to ensure more accurate and clear information. After applying data analysis techniques to the obtained data, interpretations can be made of the results. The correctness of the interpretations can be determined based on the results of other data analysis techniques applied to the same data. Therefore, it should be determined which method among the applied methods reached a more accurate result. The success obtained from the applied methods and the interpreted result are compared with other studies in the literature to ensure that the best result is achieved and the results are validated [50, 51].
Data preprocessing is considered a crucial step in the process of data mining and data analysis, where raw data is taken and transformed into a format that can be understood and analyzed by computers and machine learning methods. This stage is among the most important steps in

the analysis. Performing a healthy and appropriate preprocessing will significantly increase the success rate. In this stage, unstructured data in text and image formats should be cleaned and formatted before analysis. Processed data is even more important than the most powerful algorithms; machine learning models trained on poor data can be detrimental to the analysis attempted and yield meaningless results. When data is properly subjected to preprocessing and cleaned, much easier and more accurate results can be obtained.

In this study, the UNSW NB15 Dataset, consisting of data obtained from network traffic and various attack activities, has been analyzed alongside real-life modern normal scenarios. The dataset includes both normal and abnormal network activities. The total number of samples in the dataset is 2,540,044, structured as a subset containing 257,673 samples. The training set comprises 175,341 samples, while the test set contains 82,332 samples. The objective of the study was to perform an analysis of nine types of attacks, namely ”Analysis, Backdoor, DoS, Exploits, Fuzzers, Generic, Reconnaissance, Shellcode, and Worms,” along with Normal activities, and to obtain the most reliable classification methodology through various classification algorithms based on this analysis. There are 49 features in the samples of this dataset, categorized into time, flow, content, and additional features, making a total of separate groups of features.

In the study, the initial procedure involved importing the necessary libraries. The dataset was generated to address missing values, among others. Data analysis and visualization techniques were employed to gain a deeper comprehension of the characteristics by analyzing and detecting patterns in the data. Several studies were conducted to investigate the data distribution and the correlation between different features and to detect anomalies in the dataset. The dataset is an upgraded contemporary substitute for the traditional KDD dataset, which has become obsolete and contains numerous anomalies. The UNSW-NB15 dataset exhibits a higher degree of class balance than other datasets used for unauthorized entry detection. Despite being smaller and less redundant than other datasets, the UNSW-NB15 dataset is still adequate for training a model with high accuracy. Data preprocessing is the initial stage in the development of machine learning models. The data collected by the system frequently exhibits inconsistencies; preprocessing is conducted to ready the data for visualization and analysis. Data cleaning analyses were performed during the initial study, and no missing values were detected in the dataset. Every aspect was thoroughly analyzed, and normalization was deemed necessary due to the skewed data before visualization. At first, the ”id” attribute was eliminated from both datasets because of the distinct numbering in this characteristic that hindered the detection of duplicate records. Subsequently, by utilizing the Pandas library’s inherent functions, replicated data was detected and eliminated from both the training and test datasets. In addition, the dataset’s attributes that contained attack types were eliminated due to their lack of relevance for binary classification problems.

The encoded variables and extracted features were examined; initially, relationships among features were deduced since they only increased the dimensionality of the dataset. Before proceeding to the data analysis step, outliers were removed from the dataset to understand better how features were distributed. The PyCaret library creates a sequential arrangement consisting of all necessary function blocks or modules that can simplify the model training process. This includes the data preprocessing stage as well. Functions related to data preprocessing and preparation are important components of the library. As a result, the PyCaret library can automatically handle these functions. As a developer, the task is to call the setup function and configure the necessary functions. To deal with missing values in the dataset, PyCaret defaults to using the mean value of the feature in the case of numerical features. If necessary, a median value can also be manually selected, but in this study, it was observed that the median value provided more suitable results than the standard value.

Normalization is performed using the ”z-score” by default in PyCaret, and the yeo-johnson method is used for transformation. Data visualization techniques have been used to better analyze and identify data patterns to understand the features. Based on predictions, decisions have been made for subsequent stages regarding which optimal features to include in the dataset during training or which relevant features can be removed to improve the model’s accuracy. The Apriori algorithm created association rules for the given dataset, and sorting was done based on confidence and lift values. After sorting, each row has been duplicated to include antecedents and consequents. Following this process, sorting was done again to create a list. Frequent item columns were identified for the entire dataset, and the Apriori algorithm was again used to find subsets of frequently used items. The rule was created from subsets, and each frequent column was added for adjustment. A repeated sequence was created to find all possible columns in the dataset. The One Hot Encoding technique was applied to each nominal field in the dataset. The reason for employing this approach is that machine learning models are incapable of cannot handling our dataset, as they can only process numerical values. Therefore, it is necessary to preprocess categorical features for machine learning models. One Hot Encoding is a frequently employed technique for preprocessing categorical attributes in machine learning models. This encoding method generates a distinct binary attribute for every potential category and assigns a value of ”1” to the attribute corresponding to each instance’s original category. Converting categorical data into numerical form is a crucial step in feature engineering during the training process of learning techniques. Encoding is typically utilized to represent data in integer form. The integer-coded variable is eliminated, and a distinct binary variable is introduced for every unique integer value. A column containing encoded categorical data with labels is obtained and split into multiple columns. The numbers are substituted with random ”1”s and ”0”s according to the values assigned to each column. Although this method may benefit certain situations with a clear order among category values, it may encounter difficulties and result in poor performance when applied to input data that lacks any inherent order. Although the ”Label Encoding” method performs a similar process, it is recommended to use this method when categories are ordered and uniform. Therefore, the ”One Hot Encoding” encoding method was used, as it showed lower performance in the previous classification. Subsequently, all important parameters and objects were saved to disk to apply the same process to test data. All methods were applied to the test dataset as well, and it was considered that the preprocessing stage was completed. The final step of data preprocessing is feature scaling, which transforms all numerical values of the dataset to a standard scale. The feature scaling process was performed using the MinMaxScaler and StandardScaler functions in Scikit-Learn.

3.1. Classification

3.1.1. Classification with logistic regression

After all the preprocessing steps mentioned above, training the data primarily started with LR classification. LR is a supervised learning algorithm that predicts a dependent categorical target variable. It is a method commonly used for binary classification problems (problems with two class values). Hyperparameters such as ”alpha” and ”penalty” were set for LR. The feature scaling method for each ML algorithm is selected based on performance metrics after a comparative analysis. LR is advantageous for large datasets that need to be categorized; therefore, the LR method was initially used for classification. This method utilized the Sigmoid function, a mathematical function that maps predicted values to probabilities. The Sigmoid function maps a real value to another value in the range of ”0 and 1,” as regression values

must be between 0 and 1, creating a curve like an ”S” shape. The S-shaped curve can be called the Sigmoid or logistic function. A threshold value defining the probabilities of 0 or 1 was used in LR. Values above the threshold are biased towards 1, while values below the threshold are biased towards 0.
The dataset had been properly prepared for these operations and then trained using the training set. The Logistic Regression class from the Sklearn library was imported to provide training or fit the model to the training set. Once the class was imported, a classifier object was instantiated and utilized to train the model using logistic regression. The model was effectively trained on the training set, resulting in accurate predictions on the test set data. Subsequently, a confusion matrix (CM) was generated to evaluate the precision of the classification. This was created using the confusion matrix function from the Sklearn library. Once the function was imported, it was invoked using a fresh variable. The function requires two parameters: the real values and the predicted values generated by the classifier. The Matplotlib library was utilized to represent the outcome graphically. The LR training set result was effectively visualized to distinguish between attack and non-attack data in this classification task.

3.1.2. Classification with extra trees classifier

XT can be evaluated similarly to RF in terms of creating multiple trees and separating nodes using random subsets of features, but there are two important differences. Firstly, it does not bootstrap observations; nodes are split into random splits rather than best splits. Initially, multiple trees are created by default, without replacement nodes, and all observations are divided into random splits based on random splits between selected features in a random subset. Parameters such as max depth, min_samples_split, and min_samples_leaf are adjusted for classification. Subsequently, inputs are provided for cross-validation. After determining the number of trees to be constructed, the classification model is built, and classification is performed.

3.1.3. Classification with support vector machines

SVM is a supervised machine learning algorithm that can be used for classification and regression problems, although it is typically used for classification. When given data with 2 or more labeled data classes, it acts as a discriminative classifier defined formally by an optimal hyperplane that separates all classes. Subsequently, new examples mapped to the same space can be categorized based on which side of the gap they fall into. In SVM, support vectors are the closest data points to the hyperplane, which are points that would change the position of the hyperplane dividing the dataset. Therefore, they can be considered critical elements of a dataset. This geometry indicates that the hyperplane is a flat subset of one dimension less than the surrounding space. For instance, the hyperplane of an ndimensional space is a flat subset of dimension n-1, inherently dividing the area into two half-spaces. To train our classifier, we used a function called ”Hinge Loss,” known as ”Margin Loss.” Hinge loss is specifically used for ”maximum margin” classification, particularly for SVMs. The regularization balances between maximizing the margin and minimizing the loss. The goal is to find the decision surface that is maximally distant from any data point. The objective is to minimize and optimize the loss to learn the weights. The objective function is derived to obtain gradients, using the total rule to differentiate each term separately. The aim here is to update the weight vector using gradients of both terms if there’s a misclassified example; otherwise, the weight vector is updated only with the regularization’s gradient if the classification is correct. The learning rate determines the length of steps the algorithm takes down the gradient on the error curve. It’s important to note that the algorithm should not overshoot the optimal point, the learning rate shouldn’t be too high, and convergence shouldn’t take too long. The regularization controls the balance between achieving low training error and having the ability to generalize your classifier to unseen data with low test error. A regularization parameter of ”1/epochs” was chosen, so this parameter decreases as the number of epochs increases. Additionally, the height of the regularization is considered to be the error.

3.1.4. Classification with random forest classifier

RF can be characterized as a classifier built on decision trees. Each tree in RF makes a class prediction, and the class with the most similarity becomes the model’s prediction. In the study, estimators called predictors were initially set and assigned to an array. Then, cross-validation was performed, and their visualization was ensured. Next, the maximum depth setting and minimum sample split were performed. The purity of the subset is calculated when ”Gini” is randomly selected, quantifying the probability amount of misclassified specific features. It can be called pure if all elements are solely connected to a single class. The Gini Index criterion includes entropy as a factor, resulting in values between 0 and 1. A value of 0 indicates perfect classification purity, where all elements belong to a single class. The value ”1” denotes the stochastic allocation of elements across various categories. A Gini Index value of 0.5 indicates a balanced distribution of elements across different classes. Features with the lowest Gini Index values were prioritized when generating the decision tree. As a result, several parameters were modified with appropriate values for the classifier. The performance is thought to rely less on ”n_estimators,” ”max_depth,” and other parameters. We selected the optimal parameters for our model and performed the classification accordingly.

3.1.5. Classification with decision tree classifier

Decision trees (DT) are non-parametric supervised learning techniques that can be used for both classification and regression tasks. Decision Trees (DT) aims to construct a model that accurately predicts the value of a target variable. This is achieved by extracting simple decision rules from the features of the data. The root node in classification serves as the starting point for the decision tree. It represents the entire dataset divided into two or more homogeneous clusters. On the other hand, leaf nodes are the final output nodes, and once a leaf node is reached, the tree cannot be further divided. The process involves splitting, where the decision node/root node is divided into sub-nodes based on given conditions. A new subtree is created with the split of the tree. Pruning is performed to remove unwanted branches from the tree. The deeper the tree, the more complex the decision rules, and the more suitable the model. The tree construction began with the root node, creating a tree that includes the entire dataset. Using feature selection, the best feature in the dataset was identified, and a decision tree node containing the best attributes’ possible values was created by splitting into subsets containing the best attributes. New decision trees were iteratively created using the subsets of the generated dataset; then parameters were set for ”max_depth,” ”min_samples_split,” and ”min_samples_leaf” in classification. Subsequently, inputs were provided for cross-validation. After designing the number of trees to be

built, a classification model was created, and classification was performed. Following these processes, the performance rate was compared with studies in the literature and found to be lower. Therefore, the idea of conducting correlation analysis was considered to improve the performance. Correlation analysis is a statistical method used to evaluate the strength of the relationship between two variables that are measured on a continuous scale. This analysis method is employed to ascertain the potential correlations between variables. Correlation analysis is essential in the study because it accounts for the potential influence of unmeasured variables on the research outcomes, which is often misunderstood. A correlation between two variables indicates that any systematic change in one variable will also result in a systematic change in the other; thus, the variables can change in tandem. The magnitude of the correlation, whether positive or negative, is contingent upon the measured numerical values. The positive correlation is observed when there is a simultaneous increase in both variables; one variable's high values correlate with the other's high values. When one variable decreases and the other increases, a negative correlation exists; that is, the high values of one variable are correlated with the low values of the other. Columns in the dataset were analyzed, and columns with high correlation were removed. Initially, a value of 0.97 was considered a limit in the created correlation matrix, and columns with relationships below this value were not included in the analysis stage. The columns with the most significant relationship among them are provided in Table 3.1.

Table 3.1: Columns with the highest correlation in the dataset

COLUMN NAMES					
'spkts'	'dpkts'	'dpkts'	'sbytes'	'dbytes'	'swin'
'sloss'	'dbytes'	'dloss'	'sloss'	dloss'	'dwin'
'ct_srv_src'	'ct_dst_src_ltm'	'is_ftp_login'	'ct_srv_dst'	'ct_srv_dst'	'ct_ftp_cmd'

The data types of the fields are given in Table 3.1, and all other fields were extracted in the study. This aimed to avoid type errors when analyzing different data types simultaneously. The fields in the dataset are divided into "Nominal" and "Numeric." Nominal columns, which are non-numeric columns, are seen as proto, service, state, and attack_cat. In the preprocessing steps, symbolic features such as protocol, service, flags, etc., were processed, and all symbolic and non-numeric features that do not actively contribute to attack detection and do not require processing were removed. To improve performance, nominal attributes were converted to numeric attributes. Subsequently, an advanced forward-selected wrapper selection method was used to improve performance and reduce features. The class values of the UNSW B15 dataset were transformed from numeric to nominal form following each of these stages. The Weka data mining tool offered a numeric-to-nominal filter to convert numeric values to nominal ones. In practice, the attack class is appended as the last attribute to the dataset. On the contrary, attribute number 45 of the UNSW NB15 dataset comprises class values expressed as numeric values. In Weka, the objective of each experimental group is to investigate one particular aspect of the performance of machine learning-based intrusion detection in the presence of noise. In the first experimental group, a baseline is established to assess the potential effects of noise on machine learning-based intrusion detection systems (IDSs) and evaluate the efficacy of noise-filtering methods. Moreover, it is employed to determine the effectiveness of ensemble learning algorithms. Compared to the outcomes of subsequent stages, this stage's results are evaluated. An algorithm will filter the datasets in the second experimental set for interquartile noise. By applying this filter to the datasets, outliers, and extreme values can be identified. A comparison will be made between the outcomes of this phase and the initial target. Following that, datasets containing unauthorized entries were injected with noise in varying proportions. The noisy data was subsequently contrasted with the baseline acquired during the initial phase of noise introduction. Noise filtering was executed through the exclusion of noisy instances and the determination of varying levels of noise. This will facilitate the analysis of the impact of noise on machine learning algorithms when executed on datasets containing unauthorized entries devoid of outliers and extreme value instances. Furthermore, before noise filtration, noise injection was implemented. Noise is introduced into this experiment by manipulating a specific proportion of the training set's labels. Outliers and extreme values are eliminated to filter out noise. Finally, the effect of ensemble learning techniques applied to unauthorized entry datasets on the precision of ML algorithms was investigated. The general flow diagram of the proposed system design is presented in Figure 3.2.

In this study, initially, analyses for data cleansing were conducted, examining the description of each feature, and normalization was ensured before visualization due to the skewness of the data. Duplicate records were removed, and subsequently, attributes containing attack types in the dataset were eliminated as they were unnecessary for binary classification problems. To deal with missing values in the dataset, PyCaret defaults to using the mean value of the feature, particularly for numerical attributes. It has been observed that using the median value yields more suitable results compared to the standard value. Therefore, normalization and transformation processes have been conducted. To observe relationships in the dataset, data was visualized, association rules were generated using the Apriori algorithm, and sorting was performed based on confidence and lift. The conceptual framework is depicted in the figure. In the structure shown in the figure, the first area encompasses normal and abnormal network activities prepared by the IXIA PerfectStorm tool at the University of New South Wales in Australia. After a series of small steps, each nominal field in the dataset was encoded using the "One Hot Encoding" method. Later, in order to apply the same process to the test data, all significant parameters and objects were saved to disk. Feature scaling was performed. Subsequently, classification was carried out using LR, XT, SVM, RF, and DT methods. Then, analyses and experiments were conducted again using the KNN, BayesNet, NB, DT, and RF algorithms with both preprocessed data and the entire dataset combined using a technique called "Cross Validation."

Noise Injection: Noise can manifest differently, including incorrectly labeled data or inaccurately classified examples. To obtain more precise outcomes, it is advisable to eliminate any noisy data, as utilizing clean data assists in preventing potential problems such as overfitting. The "AddNoise filter" in Weka introduces noise into data sets. The default value for noise injection in the AddNoise filter is 10

The study is fundamentally divided into two parts. The first step was conducted on Google Colab, followed by various experimental works on Weka. Applications in Weka are generally divided into two main parts. The first part involves working on the entire dataset, while the second part entails taking pre-processed data from Colab and subjecting it to analysis again using the abovementioned steps.

Figure 3.3 shows a general block diagram showing the overall steps and algorithmic overview of two parts of the experimental studies.

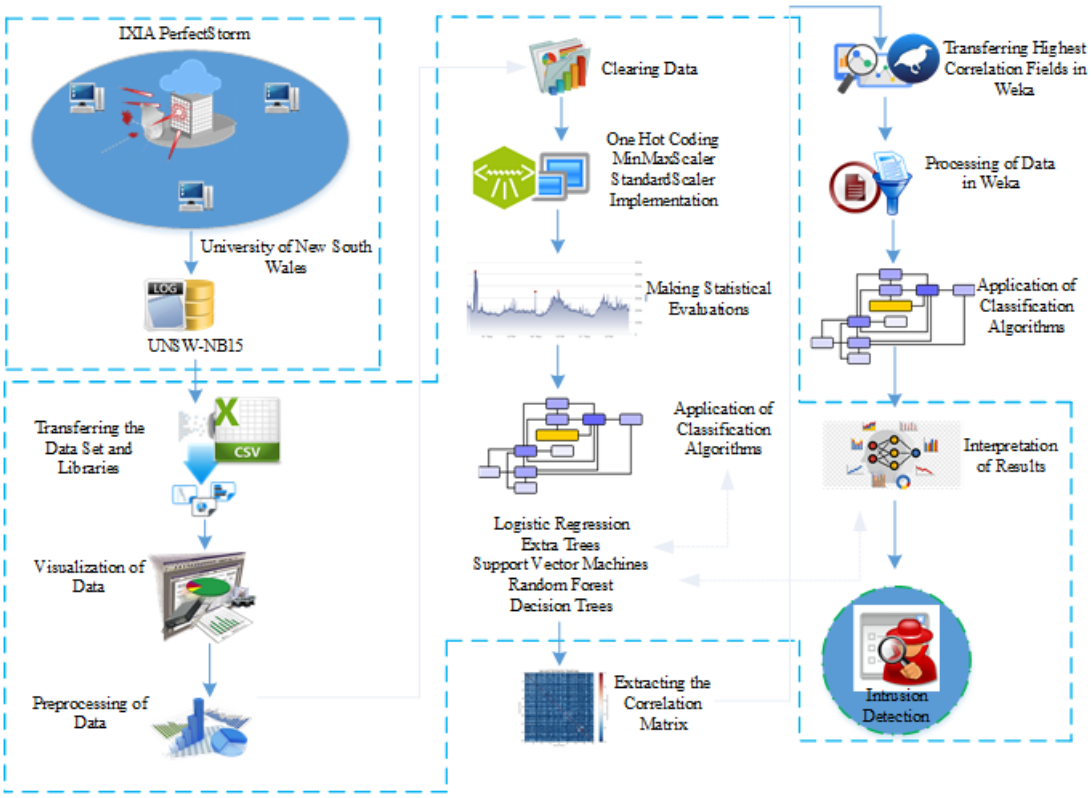


Figure 3.2: General flow diagram of the proposed system design

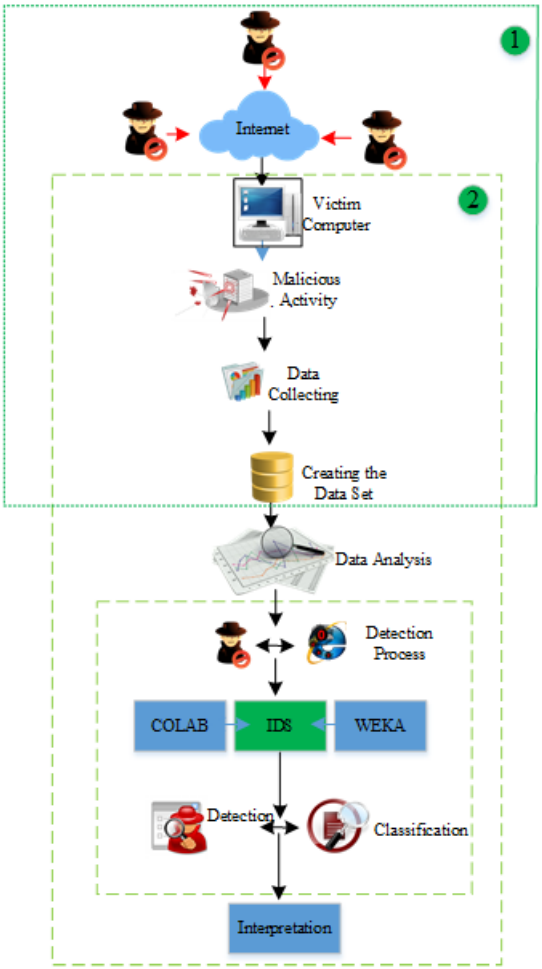


Figure 3.3: Block diagram of experimental studies on WEKA and Colab Platforms

The main steps of experimental studies conducted in Weka are as follows:

1. Loading the training data set,
2. Converting label values from numeric to nominal,
3. Training the ML algorithm on the training data set,
4. Running the ML algorithm on the test data set,
5. Reverting to step 3 using other ML algorithms,
6. Removing noise by manipulating the labels of training examples,
7. Training the ML algorithm on the noisy training data set,
8. Running the ML algorithm on the test data set,
9. Repeating step 3 until each ML algorithm is trained and tested with different noise levels,
10. Loading the unauthorized entry data set for the training section,
11. Converting label values from numeric to nominal,
12. Performing noise filtering by removing outliers and extreme values,
13. Training the ML algorithm on the noisy training data set,
14. Running the ML algorithm on the test data set,
15. Initiating ensemble learning methods (bagging-boosting),
16. Using a base classifier as an ML algorithm,
17. Training the ensemble of classifiers on the data set,
18. Testing the ensemble of classifiers on the test data set,
19. Repeating the steps until each ML algorithm is used with both ensemble learning methods.

The first stage involves conducting studies on the entire dataset, while the second stage entails retrieving preprocessed data with Colab and subjecting it to the aforementioned steps for reanalysis. Initially, analyses were conducted for data cleansing, with each feature's description examined. Due to data skewness, normalization was ensured before visualization. Duplicate records were removed, and subsequently, attributes containing attack types in the dataset were eliminated as they were unnecessary for binary classification problems. To handle missing values in the dataset, PyCaret defaults to using the mean value of the feature, particularly for numerical attributes. It was observed that the median value provided more suitable results than the standard value. Normalization and transformation processes were then carried out.

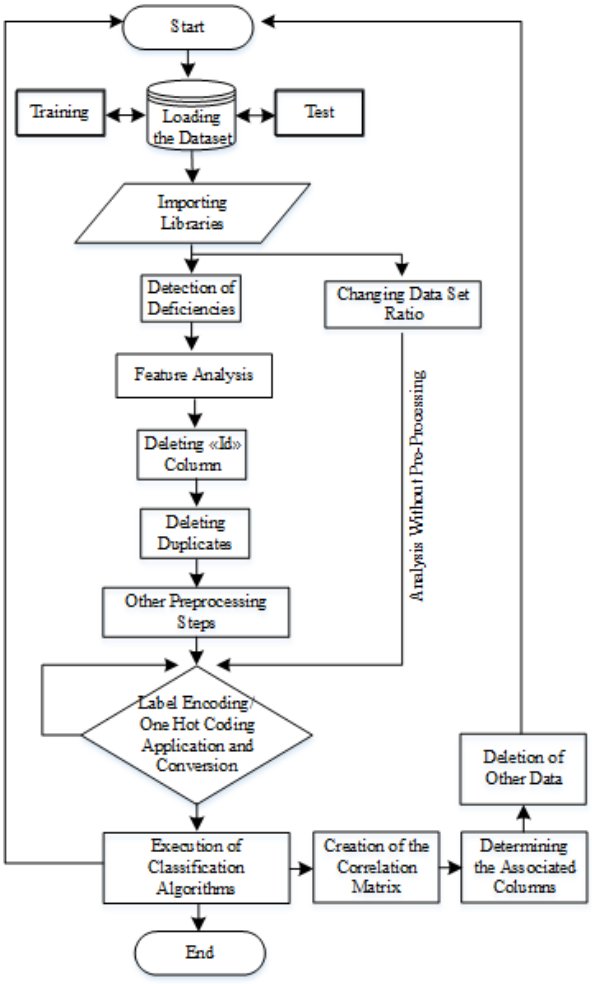


Figure 3.4: Flowchart of the experimental studies on the Colab platform

Data was visualized to observe relationships in the dataset. The Apriori algorithm generated association rules, and sorting was performed based on confidence and lift. After a series of steps, each nominal field in the dataset was encoded using the ”One Hot Encoding” method. Subsequently, all significant parameters and objects were saved to disk to apply the same process to test data. Feature scaling was conducted, followed by classification. Further analyses and experiments were conducted by combining both preprocessed data and the entire dataset using a technique called ”Cross Validation” with various algorithms. This resembles a security analytics component that detects anomalies in data to obtain intelligence for previously unseen threats. The managed rules component enables organizations to respond to the latest attacker techniques almost in real-time with nearly real-time updates for analysts. Figure 3.4 represents the flow of experimental studies on the Colab platform.

4. Results

Log files are the most important data source for ensuring network control. Therefore, the UNSW NB15 Dataset was used, which consists of data obtained from network traffic and various attack activities, in addition to real-life modern scenarios in log files. This dataset contains both normal and abnormal network activities, with 2,540,044 samples. This study used a subset of the dataset containing 257,673 samples, divided into training and test datasets. The training set comprises 175,341 samples, while the test set contains 82,332 samples, encompassing attacks such as Analysis, Backdoor, DoS, Exploits, Fuzzers, Generic, Reconnaissance, Shellcode, and Worms, along with normal activities. The environment used to run the application was the free Google Colab version, which utilizes a Tesla K80 GPU.

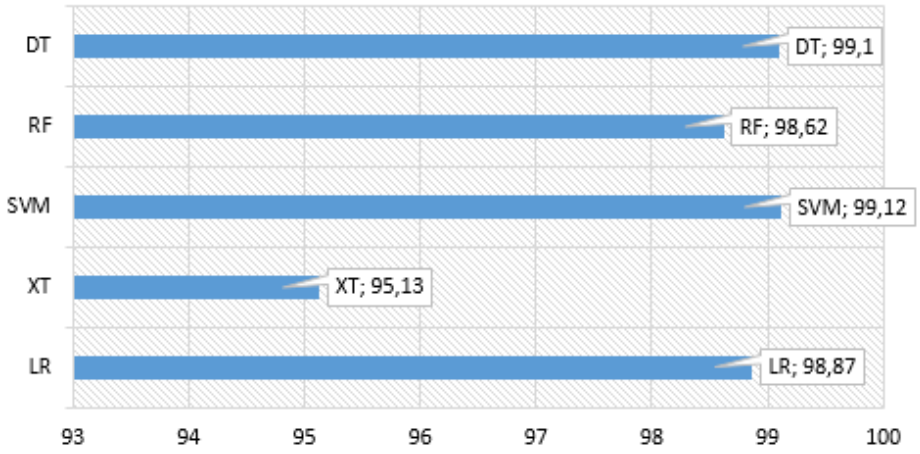


Figure 4.1: Colab Application Classification Results

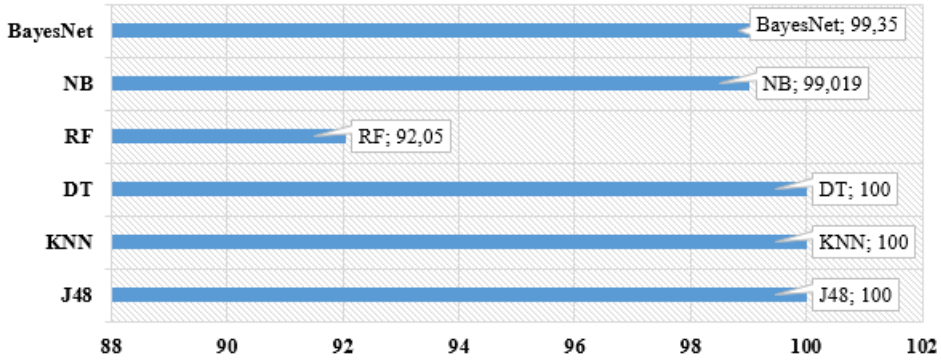


Figure 4.2: Weka Analysis Classification Results

Figure 4.1 presents the results obtained from the application conducted on Colab. Figure 4.2 displays the results of the experiments and applications conducted on Weka. As seen in Table 4.1, the collective test results of the applications show that RF yielded the lowest performance rate at 92.05% in operations performed with Weka. The highest performance was achieved by analyzing DT, J45, and KNN methods in Weka. The highest performance in the Colab application was achieved with the DT method, while the lowest performance was obtained with the implementation of the XT method. Compared to others, SVM in the Colab application produced a higher accuracy score. The PyCaret library facilitated this research to be conducted swiftly and with less effort when it became available. PyCaret’s tuning function also contributed to the study.

Table 4.1: Overall test results for applications

Colab Application Results		Weka Application Results	
Algorithm	**Accuracy(%)**	**Algorithm**	**Accuracy(%)**
RF	98,62	***RF***	92,05
DT	99,10	***DT***	100
LR	98,87	***KNN***	100
XT	95,13	***NB***	99,19
SVM	99,12	***BayesNet***	99,35

The attacks in the UNSW NB15 dataset are highly imbalanced, with some having a probability close to zero. The imbalanced distribution of data has a negative effect on the accuracy of classifying the least frequent attack types, namely Analysis, Backdoor, Shellcode, and Worms, which collectively account for a mere 2.08% of the dataset. Given that these attack classes make up over 68% of the dataset, it is possible to achieve a notably high level of performance in Exploits and Generic classes. The proposed model has demonstrated superior performance to all other Weka models, except for RF. Overall performance experiences a substantial decline when the machine learning detection and classification stages are integrated, and a semi-dynamic hyperparameter tuning strategy demonstrated substantial enhancements in multi-class models with pertinent characteristics and prevalent cybersecurity vulnerabilities. The suggested deep learning classification framework, except for the dynamic hyperparameter tuning strategy, demonstrated substantial enhancements in multi-class models compared to similar deep learning-based network intrusion detection systems (IDSs). Several experiments were conducted to determine the best-performing ML algorithm for each dataset. Additionally, it was found that the performance metrics of the training set results, including accuracy, precision, recall, F1-score, ROC, and CM test results, were higher than the baseline and in the same ranking order. Overall, the models correctly predicted 'normal' data while exhibiting misclassifications when predicting unauthorized entries. An association rule-based technique was applied for feature selection, using the point model selection for each feature. This reduced processing time for determining frequent values and identified the top-ranked features by removing irrelevant or noisy ones, resulting in the input for the machine learning model. The associated cost with false negatives should be very low in this scenario.

Upon examining these results, it is evident that data preprocessing is the first and most crucial step in machine learning model development. Data collected by systems is often inconsistent, requiring preprocessing to prepare the data for visualization and analysis. Each feature was examined, and due to the skewness of the data, normalization was necessary before visualization. Initially, the 'id' attribute was removed from both datasets due to unique numbering in this feature, preventing the identification of duplicate records. This highlights the importance of such cleaning processes. Subsequently, iterative operations identified and removed significant data from the training and test datasets. The ratios in the data set were modified in both Colab and Weka, with the training and test datasets swapped, resulting in better classification results observed in the test dataset.

Encoded variables and extracted features were examined, and correlations among features were identified to reduce the dataset's dimensionality. Before proceeding to data analysis, outliers were removed from the dataset to understand better how the features are distributed. Due to the continuous increase in cyber threats and attacks, optimizing Intrusion Detection System (IDS) capabilities has become critical. ML-based IDSs emerge as novel paradigms that could be utilized to counteract misuse and anomaly attacks. However, the classification accuracy of ML-based IDSs depends on several factors. Identifying these factors aids in developing better ML-based IDSs. The results obtained from the experiments of this study clearly demonstrate that the classification accuracy of ML-based IDSs can be influenced by specific factors related to IDSs. These factors include using the dataset for training and testing, removing outliers and extreme value samples, the addition of mislabeled examples, proper preprocessing, an abundance of experimental studies, implementation of appropriate methods, and utilization of community learning techniques. These factors have various impacts on classification accuracy. In some cases, such as the positive effect of noise on accuracy, these factors often have a negative impact on classification accuracy. However, some of these factors have significantly improved the performance of certain classification algorithms when applied to the UNSW-NB15 dataset. In some cases, these factors have not affected the accuracy of ML-based validation.

5. Conclusion

The increasing importance of information and information systems security with rapidly advancing technology has elevated the necessity of taking measures in system security to the highest level. While tools used in information security systems are diverse and powerful, they may not always succeed in identifying new security vulnerabilities arising from emerging technologies. Hence, a completely secure system can never be guaranteed. Anticipating how attackers might exploit vulnerabilities before malicious attempts and ensuring security accordingly will maximize the effectiveness of the measures taken.

Critical infrastructure systems and evolving technology form the foundation of modern societies and are crucial for maintaining national welfare. These sectors, which generally encompass agriculture, water, electricity grids, transportation, communication, and various public and private sector structures, are essential for daily life. Being interconnected and interdependent complex systems, the malfunction of just one part can lead to the failure of other dependent systems, affecting the entire infrastructure. Furthermore, they play significant roles in state security, the economy, and citizens' well-being. Given these reasons, ensuring the security of critical infrastructure plays a vital role in cybersecurity, and there should be a focus on achieving complete security for such infrastructure in Turkey. Issues related to critical infrastructure protection should not be overlooked in the shadow of cybersecurity efforts, and all necessary work and collaborations must be undertaken. Cybersecurity should be considered as a comprehensive set of measures against cyber attacks. Furthermore, institutions, organizations, and users should develop tools, policies, and practices to protect their assets, and regulatory frameworks should be established for legal documents, electronic media, events, training, and security technologies through national and international cooperation. The growth of critical infrastructure, technological advancements, and cyberspace necessitates education and training. Awareness programs, seminars, conferences, and the inclusion of cybersecurity-related courses in curricula are essential. In recent years, threats to information security have become increasingly targeted towards actively used services, turning threats into a profit model for companies. Exploitation and compromise techniques have become more complex, and analyzing daily access logs and threat monitoring is crucial in information security. SIEM

(Security Information and Event Management) is utilized by many cybersecurity operations centers to enhance monitoring capabilities by correlating events from various security devices into a consolidated framework. SIEM collects security data from network devices, servers, domain controllers, etc., stores and normalizes this data to detect trends and threats, and applies analytical methods to these data. It reports security events to an incident response team, matches specific rule sets to alert and identify security issues, and serves as a data collector, monitor, and reporting system. SIEM primarily focuses on ensuring the three main concepts of information security. Considering the rapid changes in the cybersecurity landscape, organizations need solutions that can detect and respond to known and unknown security threats, which may involve using SIEM products or developing in-house custom software. In this study, first and foremost, the definition and details of critical infrastructure are presented; the aim is to create awareness of the need to take measures to protect these structures. Additionally, the relationship between critical infrastructure and SIEM, which can be considered a whole, and the concept of software engineering as necessary for technological developments are discussed; information about software engineering at national and international levels is provided. Recent studies involving critical infrastructure, cyber security policies, and SIEM are researched, and comprehensive academic and administrative works are included. In the practical part of the study, an application is developed using real-life modern scenarios and data obtained from network traffic and various attack activities from the UNSW NB15 Dataset. The developed application analyzes 10 categories, including "Analysis, Backdoor, DoS, Exploits, Fuzzers, Generic, Reconnaissance, Shellcode, and Worms," as nine types of attacks and Normal activities. The study is divided into two main parts. The first step is conducted on Colab, followed by experimental studies on Weka. The Weka applications are divided into two parts: initially working with the entire dataset and then analyzing the preprocessed data with Colab using the aforementioned steps. The results of all these experimental studies and applications are presented in Section 4. Initially, analyses were conducted for data cleansing, descriptions of each feature were examined, and normalization was ensured before visualization due to skewed data. Duplicate records were removed, and attributes containing attack types in the dataset were removed as they were unnecessary for binary classification problems. PyCaret defaults to using the mean value of a feature in case of missing values for numerical features. The median value was observed to provide better results than the standard deviation. Normalization and transformation processes were performed. Data were visualized to observe relationships, association rules were created using the apriori algorithm, and ranking was done based on confidence and lift. Each was encoded using the "One Hot Encoding" method for nominal fields in the dataset. Subsequently, all significant parameters and objects were saved to disk to apply the same process to test data. Feature scaling was performed, followed by classification using LR, XT, SVM, RF, and DT methods. Later, analyses and experiments were conducted again with KNN, BayesNet, NB, DT, and RF algorithms using preprocessed data, and the entire dataset was combined through "Cross-validation." In the application developed with Colab, RF achieved 98.62%, DT 99.10%, LR 98.87%, XT 95.13%, and SVM 99.12% accuracy. Subsequently, in Weka experiments, RF achieved 92.05%, DT 100%, KNN 100%, J48 100%, NB 99.19%, and BayesNet 99.35% accuracy. Based on all these steps and studies, governments, institutions, organizations, and users must develop tools, policies, and practices to protect their assets. Legal documents, electronic media documents, events, training, and security technologies under national/international cooperation regulations are crucial. Advancements in information technologies have emphasized logical security alongside the physical security of systems due to their impact on infrastructures and their interrelationships and dependencies. This is because critical infrastructure IT (Information Technology) systems are often controlled and monitored through computer systems. Since computer systems use the TCP/IP protocol suite, they are exposed to security risks whether connected to the internet or not. In future work, the study conducted on Colab will be developed into a system capable of real-time data analysis and attack detection. A system with intensive reporting and graphing capabilities would benefit operations personnel. Additionally, storing datasets ourselves with a system to be established would be useful for more realistic analysis and development. The planned work also includes real-time packet analysis and suspicious packet detection in the system, along with analyzing and classifying logs and different types of log files based on the SIEM logic using methods beyond classification algorithms.

Article Information

Acknowledgements: The authors would like to express their sincere thanks to the editor and the anonymous reviewers for their helpful comments and suggestions.

Author's contributions: All authors contributed equally to the writing of this paper. All authors read and approved the final manuscript.

Conflict of interest disclosure: No potential conflict of interest was declared by the authors.

Supporting/Supporting organizations: No grants were received from any public, private or non-profit organizations for this research.

Ethical approval and participant consent: It is declared that during the preparation process of this study, scientific and ethical principles were followed and all the studies benefited from are stated in the bibliography.

Plagiarism statement: This article was scanned by the plagiarism program.

References

[1] Y. Alaca, Yapay bağışıklık sistemleri ile bilgi güvenliği ve olay yönetimi geliştirilmesi, M. Sc. Thesis, Karabük University, 2018.
[2] E. Yüksel, Experimenting, threat detection and SIEM integration with custom created honeypots, M.Sc. Thesis, Ankara Yıldırım Beyazıt University, 2019.
[3] S. İşgüzar, Siber aylaklık davranışlarının bir kamu kurumu özelinde incelenmesi: log analizine dayalı bir çalışma, M. Sc. Thesis, Fırat University, 2020.
[4] F. Akgiş, Anomali tespiti için log analizi, M. Sc. Thesis, İstanbul University-Cerrahpaşa, 2021.
[5] R. Daş, M. Z. Gündüz, *Analysis of cyber-attacks in IoT-based critical infrastructures*, Int. J. Inf. Sec. Sci., **8**(4) (2020), 122-133.
[6] D. Gökçeoğlu, Güvenlik bilgileri ve olay yönetimi (SIEM)/Log korelasyon kurallarının yazılması, Ph. D. Thesis, Fırat University, 2021.
[7] H. N. Yerlikaya, Log analysis of a large scale network by using Elastic Stack, M. Sc. Thesis, Bahçeşehir University, 2020.
[8] S. Yenal, N. Akdemir, *Uluslararası ilişkilerde yeni bir kuvvet çarpani: siber savaşlar üzerine bir vaka analizi*, Cankiri Karatekin Univ. J. Inst. Soc. Sci., **11**(1) (2020), 414-450.

[9] S. Moualla, K. Khorzom, A. Jafar, *Improving the performance of machine learning-based network intrusion detection systems on the UNSW-NB15 dataset*, Comput. Intell. Neurosci., **1** (2021), 5557577.

[10] Z. Zoghi, G. Serpen, G., *UNSW-NB15 computer security dataset: Analysis through visualization*, Secur. Priv., **7**(1) (2024), e331.

[11] A. M. Aleesa, Y. Mohammed, A. A. Mohammed, N. Sahar, *Deep-intrusion detection system with enhanced UNSW-NB15 dataset based on deep learning techniques*, J. Eng. Sci. Technol, **16**(1) (2021), 711-727.

[12] G. Kocher, G. Kumar, 2021, *Analysis of machine learning algorithms with feature selection for intrusion detection using UNSWNB15 dataset*, Int. J. Netw. Secur. Appl., **13**(1) (2021).

[13] G. Mahalakshmi, E. Uma, M. Aroosiya, M. Vinitha, *Intrusion detection system using convolutional neural network on UNSW NB15 dataset*, Advances in Parallel Computing Technologies and Applications, 2021.

[14] Abdullah, F. B. Iqbal, S. Biswas, R. Urba, Performance analysis of intrusion detection systems using the PyCaret machine learning library on the UNSW-NB15 dataset, B. Sc. Thesis, Brac University, 2021.

[15] N. Sharma, N. S. Yadav, S. Sharma, 2021, *Classification of UNSW-NB15 dataset using exploratory data analysis using ensemble learning*, EAI Endorsed Trans. Ind. Netw. Intell. Syst., **8**(29) (2021), e4-e4.

[16] M. Sarhan, S. Layeghy, M. Portmann, *Towards a standard feature set for network intrusion detection system datasets,* Mob. Netw. Appl., **27**(1) (2022), 357-370.

[17] Y. Pacheco, W. Sun, *Adversarial machine learning: a comparative study on contemporary intrusion detection datasets*, Proceedings of the 7th International Conference on Information Systems Security and Privacy (ICISSP 2021), (2021), 160-171.

[18] I. F. Kilincer, F. Ertam, A. Sengur, *Machine learning methods for cyber security intrusion detection: datasets and comparative study*, Comput. Netw., **188** (2021), 107840.

[19] G. S. Kushwah, V. Ranga, *Optimized extreme learning machine for detecting DDoS attacks in cloud computing,* Computers & Security, **105** (2021), 102260.

[20] S. Roy, A. Mandal, D. Dey, *Intelligent intrusion detection system using supervised learning*, AIJR Proceedings, (2021), 25-34.

[21] M. Ahsan, R. Gomes, M. Chowdhury, K. E. Nygard, *Enhancing machine learning prediction in cybersecurity using dynamic feature selector*, J. Cybersecur. Priv., **1**(1) (2021), 199-218.

[22] T. S. Pooja, P. Shrinivasacharya, *Evaluating neural networks using bi-directional LSTM for network IDS (intrusion detection systems) in cyber security*, Global Trans. Proc., **2**(2) (2021), 448-454.

[23] S. Thirimanne, L. Jayawardana, P. Liyanaarachchi, L. Yasakethu, *Comparative algorithm analysis for machine learning based intrusion detection system*, 10th International Conference on Information and Automation for Sustainability (ICIAFS), (2021), 191-196.

[24] M. Rani, *Effective network intrusion detection by addressing class imbalance with deep neural networks multimedia tools and applications*, Multimed. Tools Appl., **81**(6)(2022), 8499-8518.

[25] M. Ozkan-Okay, Ö. Aslan, R. Eryigit, R. Samet, *SABADT: hybrid intrusion detection approach for cyber attacks identification in WLAN*, IEEE Access, **9** (2021), 157639-157653.

[26] R. Sekhar, K. Sasirekha, P. S. Raja, K. Thangavel, *A novel GPU based intrusion detection system using deep autoencoder with Fruitfly optimization*, SN Appl. Sci., **3**(6)(2021), 1-16.

[27] S. U. Yang, 2021, *Research on network malicious behavior analysis based on deep learning*, IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), (2021), 2609-2612.

[28] H. Han, H. Kim, Y. Kim, 2022, *An efficient hyperparameter control method for a network intrusion detection system based on proximal policy optimization*, Symmetry, **14**(1) (2022), 161.

[29] K. M. Al-Gethami, M. T. Al-Akhras, M. Alawairdhi, 2021, *Empirical evaluation of noise influence on supervised machine learning algorithms using intrusion detection datasets*, Secur. Commun. , **2021**(1)(2021), 8836057.

[30] A. Meliboev, J. Alikhanov, W. Kim, *Performance evaluation of deep learning based network intrusion detection system across multiple balanced and imbalanced datasets*, Electronics, **11**(4) (2022), 515.

[31] O. A. El-Sayed, S. K. Fawzy, S. H. Tolba, R. S. Salem, Y. S. Hassan, A. M. Ahmed, A. Khattab, *Deep learning framework for accurate network intrusion detection in ITSs*, 2021 International Conference on Microelectronics (ICM), (2021), 212-215.

[32] S. Kim, L. Chen, J. Kim, *Intrusion prediction using long short-term memory deep learning with UNSW-NB15*, 2021 IEEE/ACIS 6th International Conference on Big Data, Cloud Computing and Data Science (BCD), (2021), 53-59.

[33] Z. Hossain, M. M. R. Sourov, M. Khan, P. Rahman, *Network intrusion detection using machine learning approaches*, Fifth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud), (2021), 438-442.

[34] I. Dutt, *Pre-processing of KDD'99 & UNSW-NB network intrusion datasets*, Turk. J. Comput. Math. Educ., **12**(11) (2021), 1762-1776.

[35] S. Kim, L. Chen, J. Kim, *Intrusion Prediction using LSTM and GRU with UNSW-NB15*, 2021 Computing, Communications and IoT Applications (ComComAp), (2021), 101-106.

[36] R. Singh, G. Srivastav, G., *Novel framework for anomaly detection using machine learning technique on CIC-IDS2017 dataset*, 2021 International Conference on Technological Advancements and Innovations (ICTAI), (2021), 632-636.

[37] J. V. V. Silva, N. R. de Oliveira, D. S. Medeiros, M. A. Lopez, D. M. Mattos, *A statistical analysis of intrinsic bias of network security datasets for training machine learning mechanisms*, Ann. Telecommun., **77**(7) (2022), 555-571.

[38] S. Priya, 2021, *Performance analysis comparison on various cyber-attack dataset by relatıng a deep belief network model on an intrusion detectıon system (IDS)*, Inf. Technol. Ind., **9**(3) (2021), 608-613.

[39] J. Man, G. Sun, *A residual learning-based network intrusion detection system*, Secur. Commun. Netw., **2021**(1) (2021), 5593435.

[40] L. Ashiku, C. Dagli, *Network intrusion detection system using deep learning*, Procedia Computer Science, **185** (2021), 239-247.

[41] M. K. Hooshmand, D. Hosahalli, *Network anomaly detection using deep learning techniques*, CAAI Trans. Intell. Technol., **7**(2) (2022), 228-243.

[42] I. E. Kamarudin, M. F. Ab Razak, A. Firdaus, M. I. Jaya, Y. T. Dun, *Performance Analysis on Denial of Service attack using UNSW-NB15 Dataset*, 2021 International Conference on Software Engineering & Computer Systems and 4th International Conference on Computational Science and Information Management (ICSECS-ICOCSIM), (2021), 423-426.

[43] R. Magán-Carrión, D. Urda, I. Díaz-Cano, B. Dorronsoro, *Improving the reliability of network intrusion detection systems through sataset integration*, IEEE Trans. Emerg. Top. Comput., **10**(4) (2022), 1717-1732.

[44] N. Sharma, S. Yadav, *Ensemble learning based classification of UNSW-NB15 dataset using exploratory data analysis,* 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), (2021), 1-7.

[45] Y. J. Chew, N. Lee, S. Y. Ooi, K. S. Wong, Y. H. Pang, *Benchmarking full version of GureKDDCup, UNSW-NB15, and CIDDS-001 NIDS datasets using rolling-origin resampling*, Inf. Secur. J. Global Perspect., **31**(5) (2022), 544-565.

[46] T. Acharya, I. Khatri, A. Annamalai, M. F. Chouikha, *Efficacy of machine learning-based classifiers for binary and multi-class network intrusion detection*, 2021 IEEE International Conference on Automatic Control & Intelligent Systems (I2CACIS), (2021), 402-407.

[47] G. Dlamini, M. Fahim, *DGM: a data generative model to improve minority class presence in anomaly detection domain*, Neural Comput. Appl., **33**(20) (2021), 13635-13646.

[48] A. Pavlov, N. Voloshina, *Dataset selection for attacker group identification methods*, 30th Conference of Open Innovations Association FRUCT, (2021), 171-176.

[49] H. Güler, Ö. Alpay, *Intrusion detection and classification based on deep learning*, 2021 International Conference on Information Security and Cryptology (ISCTURKIYE), (2021), 40-44.

[50] U. Gürtürk, M. Baykara, M. Karabatak, *Identifying the visitors with data mining methods from web log files*, Int. J. Emerg. Technol. Eng. Res., **5**(3) (2017), 243-249.

[51] U. Gürtürk, Türkiye'nin siber güvenlik politikalarının yazılım mühendisliği açısından değerlendirilmesi ve kritik altyapıların siber saldırılardan korunmasına yönelik olay yönetim sistemi tasarımı, M.Sc. Thesis, İstanbul University-Cerrahpaşa, 2022.

Journal of Mathematical Sciences and Modelling, 7(2) (2024) 75-81
Research Article

Journal of Mathematical Sciences and Modelling

Journal Homepage: www.dergipark.gov.tr/jmsm
ISSN 2636-8692
DOI: : https://doi.org/10.33187/jmsm.1426590

N-order Solutions to the Gardner Equation in terms of Wronskians

Pierre Gaillard

Université de Bourgogne-Franche Comté, Institut de mathématiques de Bourgogne, 9 avenue Alain Savary BP 47870, France

Article Info

Keywords: *Gardner equation, Rational Solutions, Wronskians*
2010 AMS: *33Q55, 37K10, 47.10A-, 47.35.Fg, 47.54.Bd*
Received: *27 January 2024*
Accepted: *23 June 2024*
Available online: *16 July 2024*

Abstract

N-order solutions to the Gardner equation (G) are given in terms of Wronskians of order N depending on $2N$ real parameters. We get solutions expressed with trigonometric or hyperbolic functions.
When one of the parameters goes to 0, we succeed to get for each positive integer N, rational solutions as a quotient of polynomials in x and t depending on $2N$ real parameters. We construct explicit expressions of these rational solutions for the first orders.

1. Introduction

Gardner equation is considered in the form

$$u_t + 6u(u-2)u_x + u_{xxx} + 6u_x = 0; \tag{1.1}$$

the subscripts x and t mean partial derivatives.
Gardner [1] considered this equation in 1968 in a slightly different form. This equation describes nonlinear wave effects for example, in plasma physics [2, 3], fluid flows [4], quantum fluid dynamics [5], in dusty plasmas [6], in ocean and atmosphere [7]. It can be used to describe large-amplitude internal waves [4, 8, 9].
We can quote many methods to get solutions to this equation, as the Hirota method [10], the series expansion method [11], the mapping method [12] or the method of leading-order analysis [13].
We can also cite more recent work on this equation realised by Lin et al. [14], Ghanbari et al. [15], Cao et al., Bokaeyan et al. [16], Ankiwicz et al. [17].

Here, we obtain general solutions in terms of a quotient of two Wronskians of order N. These solutions will be called solutions of order N. They depend on $2N$ real parameters and can be expressed in terms of trigonometric or hyperbolic functions.
We construct rational solutions by considering a passage to the limit when one of the parameters goes to 0. We get rational solutions which depend on $2N$ real parameters.
We construct solutions for the first orders.

2. Solutions to the Gardner Equation of Order N in terms of Wronskians

2.1. Solutions of order N in terms of Wronskians of hyperbolic sin functions

In the following, we use the Wronskian of order N of the functions $f_1, \dots, f_N$. We recall that is the determinant denoted $W(f_1, \dots, f_N)$, defined by $\det(\partial_x^{i-1} f_j)_{1\le i\le N, 1\le j\le N}$; ∂_x^i defines the partial derivative of order i with respect to x and $\partial_x^0 f_j$ being the function f_j.
a_j, b_j are arbitrary real numbers $1 \le j \le N$. We get the result :

Email address and ORCID number: Pierre.Gaillard@u-bourgogne.fr, 0000-0002-7073-8284
Cite as "Pierre Gaillard, *N-order Solutions to the Gardner Equation in terms of Wronskians*, J. Math. Sci. Model., 7(2) (2024), 75-81"

Theorem 2.1. *Let f_j be the functions defined by*

$$f_j(X,T) = \sinh\left(\frac{1}{2}a_jX - \frac{1}{2}a_j^3T + b_j\right), \qquad 1 \le i \le N,$$

then the function u expressed as

$$u(x,t) = 1 - \partial_X \ln\left(\frac{W(\partial_X(f_1),\ldots,\partial_X(f_N))}{W(f_1,\ldots,f_N)}\right)_{|X=ix,T=-it}$$

represents a solution to the Gardner equation (1.1) which depends on 2N real parameters a_j, b_j, $1 \le j \le N$.

Proof. We have proven in [18] that the function v defined by

$$v(X,T) = \partial_X \ln\left(\frac{W(\partial_X(f_1),\ldots,\partial_x(f_N))}{W(f_1,\ldots,f_N)}\right)$$

is a solution to the (mKdV) equation (2.1)

$$u_T - 6u^2u_X + u_{XXX} = 0. \tag{2.1}$$

We deduce then that the function w expresssed as

$$w(x,t) = \partial_X \ln\left(\frac{W(\partial_X(f_1),\ldots,\partial_X(f_N))}{W(f_1,\ldots,f_N)}\right)_{|X=ix,T=-it}$$

is solution to the following equation

$$u_t + 6u^2u_x + u_{xxx} = 0.$$

It can be easily checked then that the function u expressed as

$$u(x,t) = 1 - \partial_X \ln\left(\frac{W(\partial_X(f_1),\ldots,\partial_X(f_N))}{W(f_1,\ldots,f_N)}\right)_{|X=ix,T=-it}$$

is a solution to the equation (1.1). □

2.2. Different examples of solutions to the Gardner equation using sin hyperbolic generating functions

The presence of singularities in solutions makes the representation of modules of the solutions in space (x,t) inappropriate. Thus, we will not build any solution in space (x,t).
We only give the solutions of order 1, 2 and 3 in the case of generating hyperbolic sinus functions.
Solution of order 1

Proposition 2.2. *The function u expressed as*

$$\begin{aligned} u(x,t) &= \frac{n(x,t)}{d(x,t)}, \\ n(x,t) &= 2\sinh\left(1/2\,ia_1x + 1/2\,ia_1{}^3t + b_1\right)\cosh\left(1/2\,ia_1x + 1/2\,ia_1{}^3t + b_1\right) + a_1, \\ d(x,t) &= 2\sinh\left(1/2\,ia_1x + 1/2\,ia_1{}^3t + b_1\right)\cosh\left(1/2\,ia_1x + 1/2\,ia_1{}^3t + b_1\right) \end{aligned}$$

is a solution to the Gardner equation (1.1) depending on a_1, b_1 arbitrary real parameters.

Solution of order 2

Proposition 2.3. *The function u expressed as*

$$u(x,t) = \frac{n(x,t)}{d(x,t)},$$

$n(x,t) = -2\sinh(1/2\,ia_1x + 1/2\,ia_1{}^3t + b_1)\cosh(1/2\,ia_2x + 1/2\,ia_2{}^3t + b_2)a_2{}^2\cosh(1/2\,ia_1x + 1/2\,ia_1{}^3t + b_1)\sinh(1/2\,ia_2x + 1/2\,ia_2{}^3t + b_2) + 4(\cosh(1/2\,ia_2x + 1/2\,ia_2{}^3t + b_2))^2a_2a_1(\cosh(1/2\,ia_1x + 1/2\,ia_1{}^3t + b_1))^2 - 2(\cosh(1/2\,ia_2x + 1/2\,ia_2{}^3t + b_2))^2a_2a_1 - 2(\cosh(1/2\,ia_1x + 1/2\,ia_1{}^3t + b_1))^2a_1a_2 - 2\sinh(1/2\,ia_2x + 1/2\,ia_2{}^3t + b_2)\cosh(1/2\,ia_1x + 1/2\,ia_1{}^3t + b_1)a_1{}^2\cosh(1/2\,ia_2x + 1/2\,ia_2{}^3t + b_2)\sinh(1/2\,ia_1x + 1/2\,ia_1{}^3t + b_1) + a_2{}^3\cosh(1/2\,ia_1x + 1/2\,ia_1{}^3t + b_1)\sinh(1/2\,ia_1x + 1/2\,ia_1{}^3t + b_1) - \sinh(1/2\,ia_2x + 1/2\,ia_2{}^3t + b_2)a_2{}^2a_1\cosh(1/2\,ia_2x + 1/2\,ia_2{}^3t + b_2) - \sinh(1/2\,ia_1x + 1/2\,ia_1{}^3t + b_1)a_1{}^2a_2\cosh(1/2\,ia_1x + 1/2\,ia_1{}^3t + b_1) + a_1{}^3\cosh(1/2\,ia_2x + 1/2\,ia_2{}^3t + b_2)\sinh(1/2\,ia_2x + 1/2\,ia_2{}^3t + b_2)$

$d(x,t) = 2(\sinh(1/2\,ia_1x + 1/2\,ia_1{}^3t + b_1)\cosh(1/2\,ia_2x + 1/2\,ia_2{}^3t + b_2)a_2 - \sinh(1/2\,ia_2x + 1/2\,ia_2{}^3t + b_2)\cosh(1/2\,ia_1x + 1/2\,ia_1{}^3t + b_1)a_1)(-\cosh(1/2\,ia_1x + 1/2\,ia_1{}^3t + b_1)\sinh(1/2\,ia_2x + 1/2\,ia_2{}^3t + b_2)a_2 + \cosh(1/2\,ia_2x + 1/2\,ia_2{}^3t + b_2)a_1\sinh(1/2\,ia_1x + 1/2\,ia_1{}^3t + b_1))$

is a solution to the Gardner equation (1.1) with a_1, a_2, b_1, b_2 arbitrarily real parameters.

Solution of order 3
In the case of order 3, we present solution in the particular case where $a_1 = 1$, $a_2 = 2$, $a_3 = 3$, $b_1 = 0$, $b_2 = 0$, $b_3 = 0$.

Proposition 2.4. *The function u expressed as*

$$u(x,t) = \frac{n(x,t)}{d(x,t)},$$

$n(x,t) = -150\,(\cosh(1/2\,ix + 1/2\,it))^2 - 54\,(\cosh(3/2\,ix + \frac{27}{2}\,it))^2 - 384\,(\cosh(ix + 4\,it))^2 - 180\,(\cosh(1/2\,ix + 1/2\,it))^2(\cosh(3/2\,ix + \frac{27}{2}\,it))^2 + 480\,(\cosh(1/2\,ix + 1/2\,it))^2(\cosh(ix + 4\,it))^2 + 288\,(\cosh(ix + 4\,it))^2(\cosh(3/2\,ix + \frac{27}{2}\,it))^2 + 320\,(\cosh(1/2\,ix + 1/2\,it))^2\sinh(3/2\,ix + \frac{27}{2}\,it)\cosh(3/2\,ix + \frac{27}{2}\,it)(\cosh(ix + 4\,it))^2 + 576\sinh(1/2\,ix + 1/2\,it)(\cosh(ix + 4\,it))^2\cosh(1/2\,ix + 1/2\,it)(\cosh(3/2\,ix + \frac{27}{2}\,it))^2 - 180\,(\cosh(3/2\,ix + \frac{27}{2}\,it))^2\sinh(ix + 4\,it)\cosh(ix + 4\,it)(\cosh(1/2\,ix + 1/2\,it))^2 + 300\sinh(1/2\,ix + 1/2\,it)\cosh(3/2\,ix + \frac{27}{2}\,it)\cosh(1/2\,ix + 1/2\,it)\sinh(3/2\,ix + \frac{27}{2}\,it) - 600\sinh(1/2\,ix + 1/2\,it)\cosh(ix + 4\,it)\cosh(1/2\,ix + 1/2\,it)\sinh(ix + 4\,it) - 312\cosh(3/2\,ix + \frac{27}{2}\,it)\sinh(ix + 4\,it)\sinh(3/2\,ix + \frac{27}{2}\,it)\cosh(ix + 4\,it) + 90\,(\cosh(1/2\,ix + 1/2\,it))^2\sinh(ix + 4\,it)\cosh(ix + 4\,it) + 90\,(\cosh(3/2\,ix + \frac{27}{2}\,it))^2\sinh(ix + 4\,it)\cosh(ix + 4\,it) - 160\,(\cosh(ix + 4\,it))^2\sinh(3/2\,ix + \frac{27}{2}\,it)\cosh(3/2\,ix + \frac{27}{2}\,it) - 288\sinh(1/2\,ix + 1/2\,it)(\cosh(3/2\,ix + \frac{27}{2}\,it))^2\cosh(1/2\,ix + 1/2\,it) - 288\sinh(1/2\,ix + 1/2\,it)(\cosh(ix + 4\,it))^2\cosh(1/2\,ix + 1/2\,it) - 160\,(\cosh(1/2\,ix + 1/2\,it))^2\sinh(3/2\,ix + \frac{27}{2}\,it)\cosh(3/2\,ix + \frac{27}{2}\,it) - 724\sinh(1/2\,ix + 1/2\,it)\cosh(ix + 4\,it)\sinh(3/2\,ix + \frac{27}{2}\,it)\cosh(1/2\,ix + 1/2\,it)\sinh(ix + 4\,it)\cosh(3/2\,ix + \frac{27}{2}\,it)$

$d(x,t) = 2\,(16\sinh(1/2\,ix + 1/2\,it)\cosh(ix + 4\,it)\sinh(3/2\,ix + \frac{27}{2}\,it) - 9\sinh(1/2\,ix + 1/2\,it)\cosh(3/2\,ix + \frac{27}{2}\,it)\sinh(ix + 4\,it) - 5\cosh(1/2\,ix + 1/2\,it)\sinh(ix + 4\,it)\sinh(3/2\,ix + \frac{27}{2}\,it))(16\cosh(1/2\,ix + 1/2\,it)\sinh(ix + 4\,it)\cosh(3/2\,ix + \frac{27}{2}\,it) - 9\cosh(1/2\,ix + 1/2\,it)\sinh(3/2\,ix + \frac{27}{2}\,it)\cosh(ix + 4\,it) - 5\sinh(1/2\,ix + 1/2\,it)\cosh(ix + 4\,it)\cosh(3/2\,ix + \frac{27}{2}\,it))$

is a solution to the Gardner equation (1.1).

2.3. Another solutions

We can consider other types of generating functions and get other types of solutions.

2.3.1. Solutions using hyperbolic cosine generating functions

Theorem 2.5. *Let h_j, h be the following functions*

$$h_j(X,T) = \cosh\left(\frac{1}{2}a_jX - \frac{1}{2}a_j^3T + b_j\right), \qquad \text{for } 1 \le i \le N,$$

then the function u expressed as

$$u(x,t) = 1 - \partial_X \ln\left(\frac{W(\partial_X(h_1),\ldots,\partial_X(h_N))}{W(h_1,\ldots,h_N)}\right)_{|X=ix,T=-it}$$

is a solution to the Gardner equation (1.1) with a_j, b_j $1 \le j \le N$ arbitrarily real parameters.

2.3.2. Solutions using trigonometric generating functions

Theorem 2.6. *Let g_j be the functions*

$$g_j(X,T) = \cos\left(\frac{1}{2}a_jX + \frac{1}{2}a_j^3T + b_j\right), \qquad \text{for } 1 \le i \le N,$$

then the function u expressed as

$$u(x,t) = 1 - \partial_X \ln\left(\frac{W(\partial_X(g_1),\ldots,\partial_x(g_N))}{W(g_1,\ldots,g_N)}\right)_{|X=ix;T=-it}$$

is a solution to the Gardner equation (1.1), a_j, b_j $1 \le j \le N$ being arbitrarily real numbers.

Theorem 2.7. *Let k_j be the functions*

$$k_j(X,T) = \sin\left(\frac{1}{2}a_jX + \frac{1}{2}a_j^3T + b_j\right), \qquad \text{for } 1 \le i \le N,$$

then the function u expressed as

$$u(x,t) = 1 - \partial_X \ln\left(\frac{W(\partial_X(k_1),\ldots,\partial_x(k_N)}{W(k_1,\ldots,k_N)}\right)_{|X=ix,T=-it}$$

is a solution to the Gardner equation (1.1), a_j, b_j $1 \le j \le N$ being arbitrarily real numbers.

3. Rational Solutions to the Gardner Equation

In the following, we replace all parameters a_j and b_j, $1 \leq j \leq N$ by $\hat{a}_j = \sum_{k=1}^{N} a_k (je)^{2k-1}$ and $\hat{b}_j = \sum_{k=1}^{N} b_k (je)^{2k-1}$ with e an arbitrary real parameter. We use this change of parameter to obtain rational solutions to the equation (2.1); it is enough to tend e to 0.
We get the statement:

Theorem 3.1. *Let ψ_j be the functions*

$$\psi_j(X,T,e) = \sinh\left(\frac{1}{2}\sum_{k=1}^{N} a_k (je)^{2k-1} X - \frac{1}{2}\left(\sum_{k=1}^{N} a_k (je)^{2k-1}\right)^3 T + \sum_{k=1}^{N} b_k (je)^{2k-1}\right), \quad \text{for } 1 \leq j \leq N$$

then the function u expressed as

$$u(x,t) = 1 - \lim_{e\to 0} \partial_X \ln\left(\frac{W(\partial_X(\psi_1),\ldots,\partial_x(\psi_N))}{W(\psi_1,\ldots,\psi_N)}\right)_{|X=ix,T=-it}$$

is a rational solution to the Gardner equation (1.1).

Proof. This can be easily deduced from the previous result. □

We can consider other generating function as trigonometric functions. The rational solutions of the Garner equation can also be expressed without the presence of a limit. We can give the statement:

Theorem 3.2. *Let ψ, φ_j be the functions*

$$\begin{aligned}
\psi(X,T,e) &= \sinh\left(\frac{1}{2}\left(\sum_{k=1}^{N} a_k e^{2k-1}\right) X - \frac{1}{2}\left(\sum_{k=1}^{N} a_k e^{2k-1}\right)^3 T + \sum_{k=1}^{N} b_k e^{2k-1}\right), \\
\varphi_j(X,T) &= \frac{\partial^{2j-1}\psi(X,T,0)}{\partial_{2j-1} e}, \text{ for } 1 \leq j \leq N,
\end{aligned}$$

then the function v expressed as

$$v(x,t) = 1 - \partial_X \ln\left(\frac{W(\partial_X(\varphi_1),\ldots,\partial_x(\varphi_N))}{W(\varphi_1,\ldots,\varphi_N)}\right)_{|X=ix,T=-it}$$

is a rational solution to the equation (1.1) which depend on $2N$ real numbers a_j, b_j, $1 \leq j \leq N$.

Proof. It is a consequence of the combination of the columns of the determinant of the previous theorem and the passage to the limit when e tends to 0 for each column. □

Some examples of these rational solutions are given in the rest of the text. We get rational solutions but which are singular.

3.1. Rational solutions of order 1

Proposition 3.3. *The function v expressed as*

$$v(x,t) = \frac{ia_1 x + 2b_1 + a_1}{ia_1 x + 2b_1},$$

is a rational solution to the Gardner equation (1.1), a_1, b_1 being arbitrarily real numbers.

3.2. Rational solutions of order 2

Proposition 3.4. *The function v expressed as*

$$v(x,t) = \frac{n(x,t)}{d(x,t)},$$

$n(x,t) = -{a_1}^5 x^4 + 8ia_1{}^4 b_1 x^3 + 2ia_1{}^5 x^3 + 24a_1{}^3 b_1{}^2 x^2 + 12a_1{}^4 b_1 x^2 - 12a_1{}^5 tx + 24ia_1{}^2 b_2 x - 32ia_1{}^2 x b_1{}^3 - 24ia_1{}^3 x b_1{}^2 - 24ia_1 x a_2 b_1 - 12ia_1{}^5 t - 16a_1 b_1{}^4 + 48b_1 b_2 a_1 - 48a_2 b_1{}^2 - 24b_2 a_1{}^2 + 24a_1 a_2 b_1 + 24ib_1 a_1{}^4 t - 16a_1{}^2 b_1{}^3,$

$d(x,t) = (ia_1 x + 2b_1)(ia_1{}^4 x^3 + 6a_1{}^3 b_1 x^2 - 12ia_1{}^2 x b_1{}^2 + 12ia_1{}^4 t + 24b_2 a_1 - 24a_2 b_1 - 8a_1 b_1{}^3)$

is a rational solution to the Gardner equation (1.1) depending a_1, a_2, b_1, b_2 arbitrary real numbers.

3.3. Third order rational solutions

Proposition 3.5. *The function v expressed as*

$$v(x,t)=\frac{n(x,t)}{d(x,t)},$$

$n(x,t)=-ia_1{}^{12}x^9-18a_1{}^{11}b_1x^8-3a_1{}^{12}x^8+48ia_1{}^{11}b_1x^7+144ia_1{}^{10}b_1{}^2x^7+336a_1{}^{10}b_1{}^2x^6-144b_2a_1{}^9x^6-72ia_1{}^{12}tx^6+144a_2b_1a_1{}^8x^6+672a_1{}^9b_1{}^3x^6+144ia_1{}^9b_2x^5-144ia_1{}^8x^5b_1a_2-72a_1{}^{12}tx^5-1344ia_1{}^9x^5b_1{}^3-864a_1{}^{11}b_1tx^5+1728ib_2a_1{}^8b_1x^5-2016ia_1{}^8x^5b_1{}^4-1728ia_2b_1{}^2a_1{}^7x^5+8640b_2a_1{}^7b_1{}^2x^4-1440a_1{}^7b_3x^4-3360a_1{}^8b_1{}^4x^4-4320a_1{}^5a_2{}^2b_1x^4+4320a_1{}^6a_2b_2x^4+720ia_1{}^{11}tb_1x^4-8640a_2b_1{}^3a_1{}^6x^4+1440a_1{}^8b_1b_2x^4-4032a_1{}^7b_1{}^5x^4+4320ia_1{}^{10}tb_1{}^2x^4-1440a_1{}^7b_1{}^2a_2x^4+1440a_1{}^6a_3b_1x^4+5376ia_1{}^6b_1{}^6x^3+11520ia_1{}^6b_3b_1x^3+23040ia_2b_1{}^4a_1{}^5x^3-23040ib_2a_1{}^6x^3b_1{}^3+11520a_1{}^9b_1{}^3tx^3-2880ia_1{}^7x^3b_3+8640ia_1{}^6a_2b_2x^3-11520ia_1{}^5x^3a_3b_1{}^2-5760ia_1{}^7x^3b_1{}^2b_2-34560ib_2a_1{}^5x^3b_1a_2-8640ia_1{}^5x^3a_2{}^2b_1+2880a_1{}^{10}tb_1{}^2x^3+2880ia_1{}^6a_3b_1x^3+34560ia_2{}^2b_1{}^2a_1{}^4x^3+5760ia_1{}^6b_1{}^3a_2x^3+5376ia_1{}^7b_1{}^5x^3+34560a_2b_1{}^5a_1{}^4x^2-11520a_1{}^6b_2b_1{}^3x^2-34560a_1{}^4b_1{}^3a_3x^2+17280a_1{}^5a_3b_1{}^2x^2+17280a_1{}^5b_2a_2b_1x^2+17280a_1{}^6b_2{}^2x^2-34560b_2a_1{}^5b_1{}^4x^2-17280a_1{}^6b_3b_1x^2+103680a_2{}^2b_1{}^3a_1{}^3x^2+11520a_1{}^5a_2b_1{}^4x^2+5376a_1{}^6b_1{}^6x^2-5760ia_1{}^9x^2tb_1{}^3-4320a_1{}^{12}t^2x^2-103680b_2a_1{}^4b_1{}^2a_2x^2+34560a_1{}^5b_1{}^2b_3x^2-17280ia_1{}^8x^2ta_2b_1-34560a_1{}^4a_2{}^2b_1{}^2x^2-17280ia_1{}^8tx^2b_1{}^4+4608a_1{}^5b_1{}^7x^2+17280ia_1{}^9tb_2x^2+69120a_1{}^8b_1tb_2x-27648ia_2b_1{}^6a_1{}^3x+207360ib_2a_1{}^2a_2{}^2b_1x-3072ib_1{}^7a_1{}^5x-138240ia_2{}^2b_1{}^4a_1{}^2x+51840a_1{}^6ta_2b_2x-2304ia_1{}^4b_1{}^8x+46080ia_1{}^3b_1{}^4a_3x-51840a_1{}^5ta_2{}^2b_1x+17280ia_1{}^{11}b_1t^2x-69120a_1{}^7b_1{}^2ta_2x+17280a_1{}^6ta_3b_1x+138240ib_2a_1{}^3b_1{}^3a_2x+11520ib_1{}^4a_1{}^5b_2x+34560ib_1{}^3a_2{}^2a_1{}^3x-69120ia_1{}^5xb_1b_2{}^2+34560ia_2b_1{}^2a_3a_1{}^2x+34560ib_1{}^2a_2b_2a_1{}^4x-13824a_1{}^7b_1{}^5tx+27648ib_2a_1{}^4b_1{}^5x-103680ib_2{}^2a_1{}^3a_2x-11520ib_1{}^5a_1{}^4xa_2-34560ib_1{}^3a_3xa_1{}^4-17280a_1{}^7tb_3x-34560ib_2a_1{}^3a_3xb_1+34560ib_1{}^2b_3a_1{}^5x-34560ia_2b_1b_3a_1{}^3x-46080ia_1{}^4b_1{}^3b_3x-103680ia_2{}^3b_1{}^2xa_1+34560ib_2a_1{}^4b_3x-5760b_1{}^4a_1{}^8tx-103680b_2{}^2a_1{}^3a_2+17280b_1{}^2a_1{}^{10}t^2-23040b_1{}^4a_3a_1{}^3-69120ib_1{}^2a_1{}^7tb_2+34560b_2a_1{}^4b_3+17280ia_1{}^7tb_3-103680ia_1{}^6tb_2{}^2-69120b_1{}^2b_2{}^2a_1{}^4+2304ib_1{}^5a_1{}^7t+69120ib_1{}^3a_1{}^6ta_2-51840ia_1{}^6ta_2b_2+4608ia_1{}^6tb_1{}^6-51840a_2b_1a_1{}^8t^2+51840b_2a_1{}^9t^2+103680ia_1{}^5tb_2a_2b_1+8640ia_1{}^{12}t^3-69120b_2{}^3a_1{}^3-138240a_2{}^3b_1{}^3-512a_1{}^3b_1{}^9-768b_1{}^8a_1{}^4+34560ia_1{}^6tb_3b_1-34560ia_1{}^5ta_3b_1{}^2+51840ia_1{}^5ta_2{}^2b_1-17280ia_1{}^6ta_3b_1+69120b_1{}^3b_2a_1{}^3a_2+207360b_2a_1{}^2a_2{}^2b_1-34560b_2a_1{}^3a_3b_1+34560a_2b_1{}^2a_1{}^2a_3-34560a_2b_1b_3a_1{}^3+69120b_2a_1{}^2a_2b_1{}^4+69120b_2a_1{}^3b_3b_1-69120b_2a_1{}^2a_3b_1{}^2+207360b_2a_1a_2{}^2b_1{}^2-69120a_2b_1{}^2b_3a_1{}^2+69120a_2b_1{}^3a_3a_1+9216b_2a_1{}^3b_1{}^6-69120a_2{}^2b_1{}^5a_1-9216a_2b_1{}^7a_1{}^2-23040a_1{}^3b_1{}^4b_3+23040a_1{}^2b_1{}^5a_3-103680a_1a_2{}^3b_1{}^2+4608b_1{}^5b_2a_1{}^4-4608b_1{}^6a_2a_1{}^3+23040b_1{}^3b_3a_1{}^4$

$d(x,t)=(ia_1{}^4x^3+6a_1{}^3b_1x^2-12ia_1{}^2b_1{}^2x+12ia_1{}^4t-8a_1b_1{}^3+24b_2a_1-24a_2b_1)(-a_1{}^8x^6+12ia_1{}^7b_1x^5+60a_1{}^6b_1{}^2x^4-60a_1{}^8tx^3+120ib_2a_1{}^5x^3-160ia_1{}^5x^3b_1{}^3-120ia_1{}^4x^3b_1a_2-720a_1{}^3b_1{}^2a_2x^2+720a_1{}^4b_1b_2x^2+360ia_1{}^7tb_1x^2-240a_1{}^4b_1{}^4x^2-1440ia_3xa_1{}^2b_1-1440ia_1{}^3xb_1{}^2b_2+720a_1{}^6tb_1{}^2x-4320ib_2a_1{}^2a_2x+4320ia_2{}^2a_1b_1x+1440ib_3a_1{}^3x+1440ia_1{}^2b_1{}^3a_2x+192ia_1{}^3b_1{}^5x+960b_1{}^4a_1a_2-2880b_2a_1a_2b_1+2880b_3a_1{}^2b_1-480ia_1{}^5tb_1{}^3+720a_1{}^8t^2-2880b_2{}^2a_1{}^2-2880ia_1{}^5tb_2+64b_1{}^6a_1{}^2+5760a_2{}^2b_1{}^2-2880a_3a_1b_1{}^2+2880ia_1{}^4ta_2b_1-960b_2a_1{}^2b_1{}^3)$

is a rational solution to the Gardner equation (1.1).

3.4. Fourth order rational solutions

By choosing $a_j=b_j=j$ for $1\le j\le 4$, we obtain the following rational solution:

Proposition 3.6. *The function v expressed as*

$$v(x,t)=\frac{n(x,t)}{d(x,t)},$$

$n(x,t)=x^{16}-36ix^{15}-600x^{14}+6160ix^{13}+240tx^{13}+43680x^{12}-6840itx^{12}-89280tx^{11}-227136ix^{11}+707520itx^{10}-896896x^{10}+10080t^2x^{10}+3801600tx^9-237600it^2x^9+2745600ix^9+6589440x^8-2462400t^2x^8-14636160itx^8-12446720ix^7-41564160tx^7+14860800it^2x^7+172800t^3x^7+58060800t^2x^6+88197120itx^6-18450432x^6+21245952ix^5+14515200t^3x^5+139898880tx^5-153861120it^2x^5-96768000it^3x^4+18636800x^4-163891200itx^4+18144000t^4x^4-280627200t^2x^4-108864000it^4x^3+348364800it^2x^3-12042240ix^3-290304000t^3x^3-137871360tx^3+464486400it^3x^2-5406720x^2+282009600t^2x^2+78888960itx^2-217728000t^4x^2+387072000t^3x+1507328ix-217728000t^5x+145152000it^4x+27525120tx-134553600it^2x-4423680it-28753920t^2+653184000it^5+196608-132710400it^3$

$d(x,t)=(-x^6+12ix^5+60x^4-60tx^3-160ix^3+360itx^2-240x^2+720tx+192ix+720t^2+64-480it)(-x^{10}+20ix^9+180x^8-180tx^7-960ix^7+2520itx^6-3360x^6+15120tx^5+8064ix^5+13440x^4-50400itx^4-15360ix^3-100800tx^3-11520x^2+120960itx^2+5120ix-302400t^3x+80640tx+1024-23040it+604800it^3)$

is a rational solution to the Gardner equation (1.1).

3.5. Fifth order rational solutions

By choosing $a_j=j$ and $b_j=0$, for $1\le j\le N$, we get the following rational solution:

Proposition 3.7. *The function v expressed as*

$$v(x,t)=\frac{n(x,t)}{d(x,t)},$$

with

$$n(x,t) = ix^{25} + 55x^{24} - 1440ix^{23} - 23920x^{22} + 600itx^{22} + 283360ix^{21} + 28680tx^{21} + 2550240x^{20} - 650160itx^{20} - 18135040ix^{19} + 100800it^2x^{19} - 9307200tx^{19} + 4183200t^2x^{18} + 94483200itx^{18} - 104600320x^{18} + 498389760ix^{17} + 723945600tx^{17} - 81648000it^2x^{17} + 1987406080x^{16} + 6955200it^3x^{16} - 997315200t^2x^{16} - 4349802240itx^{16} - 6694420480ix^{15} - 21015982080tx^{15} + 227404800t^3x^{15} + 8554291200it^2x^{15} + 83051827200itx^{14} - 19170385920x^{14} - 3483648000it^3x^{14} + 54780364800t^2x^{14} + 271580467200tx^{13} - 33191424000t^3x^{13} - 271763251200it^2x^{13} + 254016000it^4x^{13} + 46860943360ix^{13} + 220147200000it^3x^{12} - 740421427200itx^{12} + 97981972480x^{12} - 1069286400000t^2x^{12} + 9398592000t^4x^{12} + 3387499315200it^2x^{11} - 1690327941120tx^{11} - 175272099840ix^{11} - 146313216000it^4x^{11} + 1077840691200t^3x^{11} - 4029574348800it^3x^{10} - 39626496000it^5x^{10} - 267776819200x^{10} + 3236183408640itx^{10} + 8719674163200t^2x^{10} - 1318851072000t^4x^{10} - 11734474752000t^3x^9 + 5189895782400tx^9 - 18317588889600it^2x^9 + 348109864960ix^9 - 533433600000t^5x^9 + 7823692800000it^4x^9 + 382763335680x^8 + 2469035520000it^5x^8 + 26900729856000it^3x^8 - 31411357286400t^2x^8 + 32591268864000t^4x^8 - 6944204390400itx^8 - 7696731340800tx^7 - 352835338240ix^7 + 365783040000it^6x^7 + 43797970944000it^2x^7 - 98712649728000it^4x^7 + 731566080000t^5x^7 + 48709140480000t^3x^7 - 269269073920x^6 + 10241925120000t^6x^6 + 6988022415360itx^6 - 69432665702400it^3x^6 + 49230603878400t^2x^6 - 2210304983040000t^4x^6 + 40967700480000it^5x^6 - 77097310617600t^3x^5 - 43966424678400it^2x^5 + 3669535457280000it^4x^5 + 5111579934720tx^5 - 92177326080000it^6x^5 + 167132528640ix^5 + 2027901173760000t^5x^5 - 512096256000000it^5x^4 - 4096770048000000t^6x^4 + 653749125120000it^3x^4 - 2939132313600itx^4 + 82239815680x^4 - 30490125926400t^2x^4 - 768144384000000it^7x^4 + 447068160000000t^4x^4 - 7681443840000000t^7x^3 - 30870077440ix^3 - 389128126464000it^4x^3 - 1279367577600tx^3 + 15834651033600it^2x^3 + 409243484160000t^3x^3 + 10241925120000000it^6x^3 - 7842388377600000t^5x^3 + 5796790272000t^2x^2 - 178362777600000it^3x^2 - 229419122688000t^4x^2 - 8304721920x^2 + 7374186086400000it^5x^2 + 1474837217280000t^6x^2 + 2765319782400000it^7x^2 + 396361728000itx^2 + 1426063360ix - 4835613081600t^3x + 821955133440000it^4x + 77888225280tx + 43016085504000000t^7x - 11470956134400000it^6x - 1334417817600it^2x + 3940702617600000t^5x - 4608866304000000it^8x - 7298088960it - 2458062028800000it^7 - 3745618329600000t^6 - 460886630400000t^8 + 117440512 + 13525843968000t^4 - 1453326336000t^2 - 92079783936000it^5 + 6143606784000it^3$$

and,

$$d(x,t) = (-x^{10} + 20ix^9 + 180x^8 - 180tx^7 - 960ix^7 + 2520itx^6 - 3360x^6 + 15120tx^5 + 8064ix^5 + 13440x^4 - 50400itx^4 - 15360ix^3 - 100800tx^3 - 11520x^2 + 120960itx^2 + 5120ix - 302400t^3x + 80640tx + 1024 - 23040it + 604800it^3)(-ix^{15} - 30x^{14} + 420ix^{13} - 420itx^{12} + 3640x^{12} - 21840ix^{11} - 10080tx^{11} - 96096x^{10} + 110880itx^{10} + 320320ix^9 - 25200it^2x^9 + 739200tx^9 - 453600t^2x^8 - 3326400itx^8 + 823680x^8 + 3628800it^2x^7 - 10644480tx^7 - 1647360ix^7 - 2116800it^3x^6 - 2562560x^6 + 24837120itx^6 + 16934400t^2x^6 - 50803200it^2x^5 + 3075072ix^5 + 42577920tx^5 - 25401600t^3x^5 + 2795520x^4 - 53222400itx^4 - 101606400t^2x^4 + 127008000it^3x^4 + 254016000it^4x^3 + 135475200it^2x^3 + 338688000t^3x^3 - 1863680ix^3 - 47308800tx^3 - 860160x^2 + 28385280itx^2 + 116121600t^2x^2 - 508032000it^3x^2 + 1524096000t^4x^2 - 3048192000it^4x - 58060800it^2x + 10321920tx + 245760ix - 406425600t^3x - 12902400t^2 + 135475200it^3 + 32768 - 2032128000t^4 - 1720320it + 1524096000it^5)$$

is a rational solution to the Gardner equation (1.1).

4. Conclusion

Two types of solutions to the Gardner equation have given in this work.
We first construct solutions in terms of Wronskians of order N depending on $2N$ real parameters with trigonometric or hyperbolic functions. Using a passage to the limit when one parameter goes to 0, we get rational solutions to the Gardner equation depending on $2N$ real parameters. So we construct very easily rational solutions.
This work presents different new representations of the solutions to the Gardner equation.
These results give also an efficient method to construct an infinite hierarchy of multi-parametric families of rational solutions to the Gardner equation as a quotient of polynomials in x and t depending on $2N$ real parameters.

Article Information

Author's contributions: The article has a single author. The author has read and approved the final manuscript.

Conflict of interest disclosure: The author declare that he has no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Supporting/Supporting organizations: No grants were received from any public, private or non-profit organizations for this research.

Ethical approval and participant consent: It is declared that during the preparation process of this study, scientific and ethical principles were followed and all the studies benefited from are stated in the bibliography.

Plagiarism statement: This article was scanned by the plagiarism program.

References

[1] R.M. Miura, C.S. Gardner, M.D. Kruskal, *Korteweg de Vries equation and generalizations. II. Existence of conservation laws and constants of motion*, J. Math. Phys., **9** (1968), 1204-1209.
[2] S. Watanabe, *Ion acoustic soliton in plasma with negative ion*, J. Phys. Soc. Japan, **53** (1984), 950-956.
[3] M.S. Ruderman, T. Talipova, E. Pelinovsky, *Dynamics of modulationally unstable ion-acoustic wavepackets in plasmas with negative ions*, J. Plasma Phys., **74** (2008), 639-656.

[4] R. Grimshaw, *Environmental Stratified Flows*, Topics in Environmental Fluid Mechanics, Kluwer, 2002.
[5] E. Demler, A. Maltsev, *Semiclassical solitons in strongly correlated systems of ultracold bosonic atoms in optical lattices*, Ann. Phys., **326** (2011), 1775-1805.
[6] A.H. Khater, A.A. Abdallah, O.H. El-Kalaawy, D.K. Callebaut, *Bäcklund transformations, a simple transformation and exact solutions for dust-acoustic solitary waves in dusty plasma consisting of cold dust particles and two-temperature isothermal ions*, Phys. Plasmas, **6** (1999), 4542-4547.
[7] R. Grimshaw, D. Pelinovsky, E. Pelinovsky, T. Talipova, *Wave group dynamics in weakly nonlinear long-wave models*, Phys. D, **159** (2001), 35-37.
[8] K.R. Helfrich, W.K. Melville, *Long nonlinear internal waves*, Annu. Rev. Fluid Mech., **38** (2006), 395-425.
[9] J.R. Apel, L.A. Ostrovsky, Y.A. Stepanyants, J.F. Lynch, *Internal solitons in the ocean and their effect on underwater sound*, J. Acoust. Soc. Am., **121** (2007), 695-722
[10] M. Wadati, *Wave propagation in nonlinear lattice III*, J. Phys. Soc. Jpn., **38** (1975), 681-686.
[11] M. Coffey *On series expansions giving closed form solutions of Korteweg de Vries like equations*, J. Appl. Math., **50**(6) (1990), 1580-1592.
[12] S.Y. Lou, L.L Chen, *Solitary wave solutions and cnoidal wave solutions to the combined KdV and mKdV equation*, Math Meth. Appl. Sci., **17** (1994), 339-347.
[13] J Zhang, *New solitary wave solution of the combined KdV and mKdV equation*, Int. Jour. Theo. Phys., **37** (1998), 1541-1546.
[14] L. Lin, S. Zhu, Y. Xu, Y. Shi, *Exact solutions of Gardner equations through tanh coth method*, Appl. Math., **7** (2016), 2374-2381.
[15] B. Ghanbari, D. Baleanu, *New solutions of Gardner's equation using two analytical methods*, Front. In Phys., **7** (2015), 1-15.
[16] M. Bokaeeyan, A. Ankiewicz, N. Akhmediev, *Bright and dark rogue internal waves, the Gardner equation approach*, Phys. Rev. E, **99** (2019), 062224-1-7.
[17] A. Ankiewicz, M. Bokaeeyan, *Integral relations for rogue wave formations of Gardner equation*, Nonlinear Dyn, **99** (2020), 2939-2944.
[18] P. Gaillard, *The mKdV equation and multi-parameters rational solutions*, Wave Motion, **100**, (2021), 102667-1-9.

Journal of Mathematical Sciences and Modelling, 7(2) (2024) 82-89
Research Article

Journal of Mathematical Sciences and Modelling

Journal Homepage: www.dergipark.gov.tr/jmsm
ISSN: 2636-8692
DOI: https://doi.org/10.33187/jmsm.1475294

A Robust Quintic Hermite Collocation Method for One-Dimensional Heat Conduction Equation

Selçuk Kutluay[1], Nuri Murat Yağmurlu[1*] and Ali Sercan Karakaş[2]

[1]*Department of Mathematics, Faculty of Science and Arts, Inonu University, Malatya, Türkiye*
[2]*Department of Mathematics, Graduate School of Natural and Applied Sciences, Inonu University, Malatya, Türkiye*
[*]*Corresponding author*

Article Info

Keywords: *Collocation finite element method, Fourier stability test, Heat conduction equation, Legendre and Chebyschev roots, Quintic hermite spline basis functions*
2010 AMS: *33C45, 35K05, 65L60*
Received: *29 April 2024*
Accepted: *24 June 2024*
Available online: *2 August 2024*

Abstract

In this work, a new robust numerical solution scheme constructed on Quintic Hermite Collocation Method (QHCM) utilizing the traditional Crank-Nicolson type approximation technique is developed for solving 1D heat conduction equation with certain initial and boundary conditions which is mostly handled as a prototype equation to support the reliability of many proposed new numerical methods. All temporal and spatial quantities in the equation are fully discretized using a usual Crank-Nicolson type finite difference approximation and a QHCM, respectively. In obtaining the present scheme, all the roots of the fourth degree Legendre and Chebyshev polynomials shifted to the unit interval are used as suitable inner collocation points. The obtained results from the developed scheme are found to be good enough and better than those from other schemes encountered in the literature. The scheme is also shown to be unconditionally stable by Fourier stability test.

1. Introduction

The considered problem in this study consists of one-dimensional heat conduction equation

$$\alpha^2 u_{xx} - u_t = 0, \quad t > 0, x \in [x_l, x_r] \tag{1.1}$$

given by the appropriate initial condition

$$u(x, t_{initial}) = f(x) \tag{1.2}$$

and homogenous Dirichlet boundary conditions

$$u(x_l, t) = u(x_r, t) = 0, \quad t > 0 \tag{1.3}$$

where $t_{initial} = 0$, $x_l = 0$, $x_r = l$, α and $f(x)$ respectively stand for the thermal diffusivity coefficient and a smooth function to be prescribed in the process of numerical calculations. The initial and boundary value (IBV) problem governed by Eqs.(1.1)-(1.3) is generally considered as one of the most outstanding Partial Differential Equations (PDEs) appearing especially in physics and engineering mathematics. The current problem is among the widely-known second order linear partial differential equation. This problem illustrates the fact that heat equation defines irreversible process and also at the same time presents a distance between previous and the next steps. Those equations generally arise in various areas of engineering and science to describe the variation of the temperature on a predefined solution domain over a given time period. For more information about the characteristics of various heat equations, the reader may refer to Refs. [1, 2] and references therein.
Since the classical heat conduction equation (1.1) with various types of IBVs is handled by many researchers as a pioneer prototype test problem in the first applications of many new numerical techniques, there exist many proposed numerical techniques in the literature to compute its approximate solutions. For example, some of their frequently used are finite difference method [3–8], finite element

Email addresses and ORCID numbers: selcuk.kutluay@inonu.edu.tr, 0000-0001-9610-504X (S. Kutluay), murat.yagmurlu@inonu.edu.tr, 0000-0003-1593-0254 (N. M. Yağmurlu), alisercan44@hotmail.com, 0000-0001-8622-1127 (A. S. Karakaş)
Cite as "S. Kutluay, N. M. Yağmurlu, A. S. Karakaş, *A qobust quintic hermite collocation method for one-dimensional heat conduction equation*, J. Math. Sci. Model., 7(2) (2024), 82-89"

method [9–14], wavelet method [15], spectral method [16]. To our knowledge, all studies constructed on finite elements method available in the literature have used one of the usual polynomial, trigonometric and exponential functions as B-spline bases. But recently, Kutluay *et al*. [17] have successfully applied collocation finite element method with cubic Hermite basis functions to generate more precise approximate numerical solutions of the heat conduction initial and boundary problem. There are also recently published and related articles with the presented method, problem and the equations such as given in Refs. [18–20] and therein. In this study, in order to obtain much more accurate and precise approximate solutions for the IBV problem, a new effective numerical scheme basically constructed on collocation finite element method using quintic Hermite spline basis functions, not the usual basis mentioned above, have been developed. The rest of the article is structured as follows: In Section 2, the solution domain of the problem is firstly divided into uniform partition in spatial and temporal directions. Then, the handled problem is fully discretized using Crank-Nicolson like difference approximation for the time quantities and QHCM utilizing the shifted roots of Chebyshev and Legendre polynomials as inner collocation points for the space quantities. Additionally, utilizing the initial and boundary conditions, the initial vector to be needed for starting the recursive scheme is also constructed. Thus, the heat problem is converted into a solvable system consisting of algebraic equations. In Section 3, it is shown that the scheme is unconditionally stable by using von-Neumann stability test for the modest values of spatial and temporal step sizes. In Section 4, the developed numerical scheme is used to obtain the approximate solutions of the experimental problem, and the computed results are compared with those of other researchers. In Section 5, this paper with a brief conclusion and future works is summarized.

2. Quintic Hermite Collocation Method

Throughout the manuscript, the 1D heat conduction equation which is widely given by Eq. (1.1) by the appropriate initial (1.2) and boundary conditions (1.3) is going to be handled. For generating the approximate numerical solutions of the problem, the finite element collocation method with quintic Hermite base functions is selected as a useful and powerful tool. To be able to apply this method, the solution domain $[x_l,x_r]\times[0,t_f]$ of the problem should be discretized. For this purpose, the spatial interval $[x_l,x_r]$ is divided into M equal width finite subintervals by means of the mesh points $x_j,\ j=1(1)M+1$ such that $x_l=x_1<x_2\cdots<x_M<x_{M+1}=x_r$ and $\Delta x=x_{j+1}-x_j$, and similarly the temporal interval $[0,t_f]$ is divided into N equal width finite subintervals by means of the mesh points $t_n,\ n=0(1)N$ such that $t_{initial}=t_0<t_1<\cdots<t_{N-1}<t_N=t_f$ and $\Delta t=t_{n+1}-t_n$. In fact, a non-uniform partition of the region could also be selected. But, since the non-uniform choice would increase the workload, computer storage capacity and running time, the uniform one has been preferred.

2.1. Discretization of space variable

In this method, the approximate solution $u_M(x,t)$ corresponding to analytical solution $u(x,t)$ will be sought by means of quintic Hermite basis functions H_{ji} as [21]

$$u(x,t)\approx u_M(x,t)=\sum_{j=1}^{M+2}a_{j+4k-4}(t)H_{ji} \tag{2.1}$$

where a's stand for the time dependent coefficients which are going to be determined later, k stands for the element number in each subinterval, $H_{4j-R}(x)$ for $R=-2(1)3$ are defined over the interval $[x_{j-1},x_{j+1}]$ as given in Ref. [22]

$$\begin{aligned}
H_{4j-3}(x) &= 6\frac{(x-x_{j-1})^5}{h^5}-15\frac{(x-x_{j-1})^4}{h^4}+10\frac{(x-x_{j-1})^3}{h^3}, && x_{j-1}\le x\le x_j\\
H_{4j+1}(x) &= 6\frac{(x_{j-1}-x)^5}{h^5}-15\frac{(x_{j-1}-x)^4}{h^4}+10\frac{(x_{j-1}-x)^3}{h^3}, && x_j\le x\le x_{j+1}\\
H_{4j-2}(x) &= 3\frac{(x-x_{j-1})^5}{h^4}-7\frac{(x-x_{j-1})^4}{h^3}+4\frac{(x-x_{j-1})^3}{h^2}, && x_{j-1}\le x\le x_j\\
H_{4j+2}(x) &= 3\frac{(x_{j-1}-x)^5}{h^4}-7\frac{(x_{j-1}-x)^4}{h^3}+4\frac{(x_{j-1}-x)^3}{h^2}, && x_j\le x\le x_{j+1}\\
H_{4j-1}(x) &= \frac{1}{2}\frac{(x-x_{j-1})^5}{h^3}-\frac{(x-x_{j-1})^4}{h^2}+\frac{1}{2}\frac{(x-x_{j-1})^3}{h^1}, && x_{j-1}\le x\le x_j\\
H_{4j}(x) &= \frac{1}{2}\frac{(x_{j-1}-x)^5}{h^3}-\frac{(x_{j-1}-x)^4}{h^2}+\frac{1}{2}\frac{(x_{j-1}-x)^3}{h}, && x_j\le x\le x_{j+1}
\end{aligned} \tag{2.2}$$

and finally i $(i=1(1)4)$ stands for the inner-collocation points to be taken from the shifted roots of Legendre and Chebyshev polynomials. Throughout this study, the following shifted roots of the fourth-degree Legendre and Chebyshev polynomials [23] computed by the symbolic programming language Matlab are respectively used as inner-collocation points

$$\begin{aligned}
\eta_1^L &= 0.069431844202974,\ \eta_2^L=0.330009478207572\\
\eta_3^L &= 0.669990521792428,\ \eta_4^L=0.930568155797026.\\
\eta_1^C &= 0.038060233744357,\ \eta_2^C=0.308658283817455\\
\eta_3^C &= 0.691341716182545,\ \eta_4^C=0.961939766255643
\end{aligned}$$

When the following local coordinate system has been utilized on the j^{th} element

$$\eta=\frac{x-x_j}{h}$$

the interval $\left[x_j,x_{j+1}\right]$ is converted into an unit interval $[0,1]$ in which η represents both Chebyshev and Legendre shifted roots. Thus, from Eq. (2.2), quintic Hermite spline functions $H_j(\eta)$ $(j=1(1)6)$ in terms of the local coordinate η are written as

$$\begin{aligned} H_1(\eta) &= 1-10\eta^3+15\eta^4-6\eta^5, & H_2(\eta) &= (\eta-6\eta^3+8\eta^4-3\eta^5)\Delta x, \\ H_3(\eta) &= \left(0.5\eta^2-1.5\eta^3+1.5\eta^4-0.5\eta^5\right)(\Delta x)^2, & H_4(\eta) &= \left(0.5\eta^3-\eta^4+0.5\eta^5\right)(\Delta x)^2, \\ H_5(\eta) &= 10\eta^3-15\eta^4+6\eta^5, & H_6(\eta) &= (-4\eta^3+7\eta^4-3\eta^5)\Delta x. \end{aligned} \tag{2.3}$$

and therefore the approximate solution $u_M(x,t)$ given by Eq. (2.1) becomes

$$u_M(\eta,t) = \sum_{j=1}^{6} a_{j+4k-4}(t) H_j(\eta). \tag{2.4}$$

From (2.3), the second order derivatives of $H_j(\eta)$ are found as follows

$$\begin{aligned} H_1''(\eta) &= -60\eta+180\eta^2-120\eta^3, & H_2''(\eta) &= \left(-36\eta+96\eta^2-60\eta^3\right)\Delta x, \\ H_3''(\eta) &= \left(1-9\eta+18\eta^2-10\eta^3\right)(\Delta x)^2, & H_4''(\eta) &= \left(3\eta-12\eta^2+10\eta^3\right)(\Delta x)^2, \\ H_5''(\eta) &= 60\eta-180\eta^2+120\eta^3, & H_6''(\eta) &= \left(-24\eta+84\eta^2-60\eta^3\right)\Delta x. \end{aligned}$$

Thus, the point wise values of the approximation (2.4) with its second order derivative at inner collocation points η_i for $i=1(1)4$ are found as follows

$$\begin{aligned} u_i &= u(\eta_i) = a_{4k-3}H_1(\eta_i)+a_{4k-2}H_2(\eta_i)+a_{4k-1}H_3(\eta_i)+a_{4k}H_4(\eta_i)+a_{4k+1}H_5(\eta_i)+a_{4k+2}H_6(\eta_i), \\ (\Delta x)^2 u_i'' &= (\Delta x)^2 u''(\eta_i) = a_{4k-3}H_1''(\eta_i)+a_{4k-2}H_2''(\eta_i)+a_{4k-1}H_3''(\eta_i)+a_{4k}H_4''(\eta_i)+a_{4k+1}H_5''(\eta_i)+a_{4k+2}H_6''(\eta_i) \end{aligned} \tag{2.5}$$

where $H_1(\eta)=H_5(1-\eta)$, $H_2(\eta)=-H_6(1-\eta)$, $H_3(\eta)=-H_4(1-\eta)$ and $H_{ji}=H_j(\eta_i)$.
In the building of the numerical scheme, the finite element method constructed on quintic Hermite spline basis functions in the discretizaion of spatial quantities and the finite difference method constructed on Crank-Nicolson like approximation in the discretization of temporal quantities will be used. This choice has the advantages of several vital characteristics such as easy to handle algorithms produced by quintic Hermite spline basis functions and low level of storage requirement. Besides those advantages, both of the non-linear and linear systems resulted from the usage of splines are usually not ill-conditioned and thus permit the required coefficients to be found out in an easy manner. Furthermore, the newly obtained approximate solutions generally don't result in numerical instability.

2.2. Discretization of time variable

Now, we are ready to discretize the aforementioned 1D heat conduction equation given by (1.1). To do this, one can use Crank-Nicolson type formula. First of all, one discretizes Eq. (1.1) as

$$\frac{u^{n+1}-u^n}{\Delta t} - \alpha^2\left[\frac{(u_{xx})^n+(u_{xx})^{n+1}}{2}\right] = 0.$$

Before proceeding more, let us separate the above equation such that the unknown values at $(n+1)$. time level are on the left-hand side and the known values n. time level are on the right-hand side as

$$\frac{u^{n+1}}{\Delta t} - \alpha^2\frac{(u_{xx})^{n+1}}{2} = \frac{u^n}{\Delta t} + \alpha^2\frac{(u_{xx})^n}{2}. \tag{2.6}$$

If one puts Eq. (2.5) in Eq. (2.6), the following fully discretized difference equation system with $4M$ difference equations and $4M+2$ coefficients in both time and space variables is obtained for the coefficients $\mathbf{a}$ to be calculated

$$\begin{aligned} & \frac{1}{\Delta t}\left[a_{4k-3}^{n+1}H_{1i}+a_{4k-2}^{n+1}H_{2i}+a_{4k-1}^{n+1}H_{3i}+a_{4k}^{n+1}H_{4i}+a_{4k+1}^{n+1}H_{5i}+a_{4k+2}^{n+1}H_{6i}\right] \\ - & \frac{\alpha^2}{2(\Delta x)^2}\left[a_{4k-3}^{n+1}B_{1i}+a_{4k-2}^{n+1}B_{2i}+a_{4k-1}^{n+1}B_{3i}+a_{4k}^{n+1}B_{4i}+a_{4k+1}^{n+1}B_{5i}+a_{4k+2}^{n+1}B_{6i}\right] \\ = & \frac{1}{\Delta t}\left[a_{4k-3}^{n}H_{1i}+a_{4k-2}^{n}H_{2i}+a_{4k-1}^{n}H_{3i}+a_{4k}^{n}H_{4i}+a_{4k+1}^{n}H_{5i}+a_{4k+2}^{n}H_{6i}\right] \\ + & \frac{\alpha^2}{2(\Delta x)^2}\left[a_{4k-3}^{n}B_{1i}+a_{4k-2}^{n}B_{2i}+a_{4k-1}^{n}B_{3i}+a_{4k}^{n}B_{4i}+a_{4k+1}^{n}B_{5i}+a_{4k+2}^{n}B_{6i}\right]. \end{aligned} \tag{2.7}$$

These newly obtained equations are clearly recursive in nature. Thus, the unknown vector $\mathbf{a}^n=(a_1^n,\ldots,a_{4M+1}^n,a_{4M+2}^n)$ can be recursively determined up to the requested final time t_f. If one utilizes the conditions given at the boundary of the solution domain by Eq. (1.3) and eliminates the coefficients a_1^n, a_{4M+1}^n in Eq. (2.7), the following statements are easily obtained. Using the boundary condition given at the left of the solution domain, the value of $u(x_l,t)$ is written as

$$u(x_l,t) = a_1^n H_{11}+a_2^n H_{21}+a_3^n H_{31}+a_4^n H_{41}+a_5^n H_{51}+a_6^n H_{61} = 0.$$

Since $H_{21}=H_{31}=H_{41}=H_{51}=H_{61}=0$ and $H_{11}\neq 0$, the value $a_1^n=0$ is found. Similarly, by the boundary condition given at the right of the solution domain, the value of $u(x_r,t)$ is obtained as

$$u(x_r,t)=a_{4M-3}^n H_{14}+a_{4M-2}^n H_{24}+a_{4M-1}^n H_{34}+a_{4M}^n H_{44}+a_{4M+1}^n H_{54}+a_{4M+2}^n H_{64}=0.$$

Since $H_{14}=H_{24}=H_{34}=H_{44}=H_{64}=0$ and $H_{54}\neq 0$, the value $a_{4M+1}^n=0$ is found.
After all of the above preparations and mathematical procedures, the iterative numerical scheme resulting in the following matrix algebraic system excluding a_1^n and a_{4M+1}^n is obtained as

$$\mathbf{L}\mathbf{a}^{n+1}=\mathbf{R}\mathbf{a}^n. \tag{2.8}$$

Here, both $\mathbf{L}$ and $\mathbf{R}$ are $4M\times 4M$ diagonal band matrices, and $\mathbf{a}^{n+1}$ and $\mathbf{a}^n$ are $4M$ column matrices.
The entries of $\mathbf{a}$ in Eq. (2.8) are obtained and then the numerical results of the 1D heat conduction equation at the next time level are calculated. This iterative process is done in a successive manner until the wanted time t_f. To be able to start this iterative calculation, the initial vector $\mathbf{a}^0$ should be found. The initial vector is determined by the usage of the initial condition given together with the governing equation. Utilizing the initial vector, Eq. (2.8) results in $4M$ unknowns. To solve the system Eq. (2.8), a script written in MatlabR2021a is used on a digital computer.

2.3. Determination of initial vector

It is obvious that there is a need for determining the initial vector to iterate the numerical scheme (2.8). This initial vector $\mathbf{a}^0$ is constructed using the conditions given at the initial time and boundaries of the solution domain. The approximate numerical solution given by Eq. (2.1) should be rewritten in terms of the initial vector as

$$u(x,t)\approx u_M(x,t)=\sum_{j=1}^{M+2} a_{j+4k-4}^0(t)\,H_{ji}.$$

Since the initial approximate solution $u_M(x,0)$ should satisfy the initial prescribed solution $u(x,0)$, it is clearly

$$u_M(x_j,0)=u(x_j,0),\qquad j=1(1)M+1.$$

From this equality, the following matrix equation is obtained

$$\mathbf{W}\mathbf{a}^0=\mathbf{b}$$

where

$$\mathbf{W}=\begin{bmatrix}
H_{21} & H_{31} & H_{41} & H_{51} & H_{61} & & & & & & & & & & \\
H_{22} & H_{32} & H_{42} & H_{52} & H_{62} & & & & & & & & & & \\
H_{23} & H_{33} & H_{43} & H_{53} & H_{63} & & & & & & & & & & \\
H_{24} & H_{34} & H_{44} & H_{54} & H_{64} & & & & & & & & & & \\
 & & & H_{11} & H_{21} & H_{31} & H_{41} & H_{51} & H_{61} & & & & & & \\
 & & & H_{12} & H_{22} & H_{32} & H_{42} & H_{52} & H_{62} & & & & & & \\
 & & & H_{13} & H_{23} & H_{33} & H_{43} & H_{53} & H_{63} & & & & & & \\
 & & & H_{14} & H_{24} & H_{34} & H_{44} & H_{54} & H_{64} & & & & & & \\
 & & & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & & & & & \\
 & & & & & H_{11} & H_{21} & H_{31} & H_{41} & H_{51} & H_{61} & & & & \\
 & & & & & H_{12} & H_{22} & H_{32} & H_{42} & H_{52} & H_{62} & & & & \\
 & & & & & H_{13} & H_{23} & H_{33} & H_{43} & H_{53} & H_{63} & & & & \\
 & & & & & H_{14} & H_{24} & H_{34} & H_{44} & H_{54} & H_{64} & & & & \\
 & & & & & & & & & H_{11} & H_{21} & H_{31} & H_{41} & H_{61} \\
 & & & & & & & & & H_{12} & H_{22} & H_{32} & H_{42} & H_{62} \\
 & & & & & & & & & H_{13} & H_{23} & H_{33} & H_{43} & H_{63} \\
 & & & & & & & & & H_{14} & H_{24} & H_{34} & H_{44} & H_{64}
\end{bmatrix},$$

$$\mathbf{a}^0=(a_2,a_3,a_4,\ldots,a_{4M-1},a_{4M},a_{4M+2})^T$$

and

$$\mathbf{b}=(u(x_{11},0),u(x_{12},0),u(x_{13},0),u(x_{14},0),\ldots,u(x_{M3},0),u(x_{M4},0))^T.$$

Therefore, from the iterative scheme (2.8), the approximate numerical solutions of the heat conduction equation given by (1.1) at the desired time are calculated.

3. Stability Analysis of the Scheme

In order to examine the stability of the linear iterative scheme in Eq. (2.7), von-Neumann method will be used. For this aim, substituting the Fourier mode

$$a_j^n=\xi^n e^{ij\varphi}$$

into Eq. (2.7), one gets

$$\begin{aligned}
&\xi^{n+1}e^{i(4j-3)\varphi}(\alpha_1)+\xi^{n+1}e^{i(4j-2)\varphi}(\alpha_2)+\xi^{n+1}e^{i(4j-1)\varphi}(\alpha_3)+\xi^{n+1}e^{i(4j)\varphi}(\alpha_4)+\xi^{n+1}e^{i(4j+1)\varphi}(\alpha_5)+\xi^{n+1}e^{i(4j+2)\varphi}(\alpha_6)\\
=\;&\xi^{n}e^{i(4j-3)\varphi}(\beta_1)+\xi^{n}e^{i(4j-2)\varphi}(\beta_2)+\xi^{n}e^{i(4j-1)\varphi}(\beta_3)+\xi^{n}e^{i(4j)\varphi}(\beta_4)+\xi^{n}e^{i(4j+1)\varphi}(\beta_5)+\xi^{n}e^{i(4j+2)\varphi}(\beta_6)
\end{aligned}$$

where $\varphi = \beta h$, β is the mode number, h is the spatial step size, $i = \sqrt{-1}$ and

$$\begin{aligned}
\alpha_1 &= H_{1i} - \tfrac{\alpha^2}{2h^2} kB_{1i} & \beta_1 &= H_{1i} + \tfrac{\alpha^2}{2h^2} kB_{1i} \\
\alpha_2 &= H_{2i} - \tfrac{\alpha^2}{2h^2} kB_{2i} & \beta_2 &= H_{2i} + \tfrac{\alpha^2}{2h^2} kB_{2i} \\
\alpha_3 &= H_{3i} - \tfrac{\alpha^2}{2h^2} kB_{3i} & \beta_3 &= H_{3i} + \tfrac{\alpha^2}{2h^2} kB_{3i} \\
\alpha_4 &= H_{4i} - \tfrac{\alpha^2}{2h^2} kB_{4i} & \beta_4 &= H_{4i} + \tfrac{\alpha^2}{2h^2} kB_{4i} \\
\alpha_5 &= H_{5i} - \tfrac{\alpha^2}{2h^2} kB_{5i} & \beta_5 &= H_{5i} + \tfrac{\alpha^2}{2h^2} kB_{5i} \\
\alpha_6 &= H_{6i} - \tfrac{\alpha^2}{2h^2} kB_{6i} & \beta_6 &= H_{6i} + \tfrac{\alpha^2}{2h^2} kB_{6i}
\end{aligned}$$

Again, if one makes the required simplification and mathematical operations, one obtains

$$\xi = \frac{M_1 - iM_2}{M_3 - iM_4} \tag{3.1}$$

where

$$\begin{aligned}
M_1 &= (\beta_3 + \beta_5)\cos\varphi + (\beta_2 + \beta_6)\cos 2\varphi + \beta_1 \cos 3\varphi + \beta_4 \\
M_2 &= (\beta_3 - \beta_5)\sin\varphi + (\beta_2 - \beta_6)\sin 2\varphi + \beta_1 \sin 3\varphi + \beta_4 \\
M_3 &= (\alpha_3 + \alpha_5)\cos\varphi + (\alpha_2 + \alpha_6)\cos 2\varphi + \alpha_1 \cos 3\varphi + \alpha_4 \\
M_4 &= (\alpha_3 - \alpha_5)\sin\varphi + (\alpha_2 - \alpha_6)\sin 2\varphi + \alpha_1 \sin 3\varphi + \alpha_4 .
\end{aligned}$$

When the modulus of Eq. (3.1) is taken, it is seen that the condition $|\xi| \leq 1$ is satisfied. Thus, it is concluded that the numerical scheme is unconditionally stable.

4. Numerical Example and Results

In this section, the obtained scheme (2.8) will be applied to the 1D heat conduction problem given by Eqs. (1.1)-(1.3) for $x \in [0,1]$, $t \in [0,t_f]$, $f(x) = \sin(\pi x)$ and $\alpha = 1$, as a test problem. The exact solution of the test problem is [7, 8]

$$u(x,t) = \sin(\pi x) e^{-\alpha^2 \pi^2 t}.$$

Since the test problem has an exact solution, the error norms L_2 and L_∞ given as follows, respectively, are going to be used to test the accuracy and validity of the scheme

$$L_2 = \sqrt{\left(h \sum_{i=1}^{M} |u_i - (u_M)_i|^2 \right)}, \qquad L_\infty = \max_{1 \leq i \leq M} |u_i - (u_M)_i| .$$

This manuscript carries out all of the numerical computations using both Quintic Hermite Collocation Method with Legendre roots (QHCM-L) and Quintic Hermite Collocation Method with Chebyshev roots (QHCM-C). All calculations have been made with MATLAB R2021a on 13th Gen Intel(R) Core(TM) i9-13900HX 2.20 GHz computer having 32,0 GB of RAM.
Some numerical results are presented for different spatial and temporal step sizes with final desired values of time to check the efficiency and accuracy of the scheme. The newly obtained results are also compared with some of the existing ones using the same parameter values.

	L_2			
Δt	QHCM-L	QHCM-C	[7]	[17]CHCM-L
0.01	5.8454×10^{-7}	5.8454×10^{-7}	4.2273×10^{-4}	4.1333×10^{-7}
0.005	1.4641×10^{-7}	1.4641×10^{-7}	1.0560×10^{-4}	1.0353×10^{-7}
0.0025	3.6621×10^{-8}	3.6621×10^{-8}	2.6395×10^{-5}	2.5895×10^{-8}

Table 4.1: A comparison of the computed error norms L_2 of the present scheme with those in Refs. [7, 17], for $M = 1000$ and $\Delta t = 0.01, 0.005, 0.0025$ $t_f = 1$).

The values of the error norm L_2 are calculated from the scheme (2.8) for the parameters $\Delta t = 1/100, 5/1000, 25/10000$ and $\Delta x = 1/1000$ at $t_f = 1$, and are listed in Table 4.1 with a comparison of those given in Refs. [7, 17]. The newly obtained error norm L_2 is remarkably small enough for using both QHCM-C and QHCM-L schemes. They are also in very good agreement with those in Ref. [17] and much better than those in Ref. [7].
In Table 4.2, a clear comparison of the newly computed error norm values L_2 of the scheme with those in Refs. [8, 17] for several values of $\Delta x = 1/5, 1/10, 1/20$ and $\Delta t = 1/10^6$ at $t_f = 1$ is displayed. It can be easily seen from the table that the obtained results are extremely small

	L_2				
Δx	QHCM-L	QHCM-C	[8](CN-I)	[8](CN-II)	[17]CHCM-L
$1/5$	3.1385×10^{-12}	2.2952×10^{-9}	4.7696×10^{-6}	8.5859×10^{-3}	3.5716×10^{-8}
$1/10$	4.7257×10^{-14}	1.5029×10^{-10}	7.7143×10^{-9}	2.1412×10^{-3}	2.2848×10^{-9}
$1/20$	1.1057×10^{-14}	9.4983×10^{-12}	1.8820×10^{-11}	5.3498×10^{-4}	1.4361×10^{-10}
$1/40$	1.3947×10^{-15}	5.9155×10^{-13}			
$1/80$	5.1467×10^{-16}	3.3364×10^{-14}			

Table 4.2: A comparison of the computed error norms L_2 of the present scheme with those in Refs. [8, 17], for various values of $M = 5, 10, 20, 40, 80$ and $\Delta t = 1/10^6$ at $t_f = 1$.

	L_2			
Δx	QHCM-L	[7]		[17]CHCM-L
		$\theta = 0.1$	$\theta = 0.2$	
$1/10$	3.5183×10^{-10}	1.6534×10^{-4}	2.4968×10^{-4}	1.6957×10^{-6}
$1/20$	8.5012×10^{-12}	4.3906×10^{-7}	7.8152×10^{-7}	1.0426×10^{-7}
$1/40$	1.0432×10^{-12}	8.6154×10^{-10}	8.0111×10^{-10}	6.4916×10^{-9}
		$\theta = 0.3$	$\theta = 0.4$	
$1/10$	3.5183×10^{-10}	3.2357×10^{-4}	3.6276×10^{-4}	1.6957×10^{-6}
$1/20$	8.5012×10^{-12}	1.0791×10^{-6}	1.2556×10^{-6}	1.0426×10^{-7}
$1/40$	1.0432×10^{-12}	9.2171×10^{-10}	1.1166×10^{-9}	6.4916×10^{-9}
		$\theta = 0.5$	$\theta = 0.6$	
$1/10$	3.5183×10^{-10}	3.5225×10^{-4}	2.8316×10^{-4}	1.6957×10^{-6}
$1/20$	8.5012×10^{-12}	1.2494×10^{-6}	1.0109×10^{-6}	1.0426×10^{-7}
$1/40$	1.0432×10^{-12}	1.2167×10^{-9}	1.1107×10^{-9}	6.4916×10^{-9}
		$\theta = 0.7$	$\theta = 0.8$	
$1/10$	3.5183×10^{-10}	1.5954×10^{-4}	1.9640×10^{-4}	1.6957×10^{-6}
$1/20$	8.5012×10^{-12}	5.4419×10^{-7}	9.1503×10^{-7}	1.0426×10^{-7}
$1/40$	1.0432×10^{-12}	7.4804×10^{-10}	8.6751×10^{-10}	6.4916×10^{-9}
		$\theta = 0.9$	$\theta = 0.95$	
$1/10$	3.5183×10^{-10}	4.5197×10^{-4}	6.1645×10^{-4}	1.6957×10^{-6}
$1/20$	8.5012×10^{-12}	2.0820×10^{-6}	2.8818×10^{-6}	1.0426×10^{-7}
$1/40$	1.0432×10^{-12}	2.4668×10^{-9}	3.6858×10^{-9}	6.4916×10^{-9}

Table 4.3: A comparison of the calculated error norms L_2 of the present scheme with those in Refs. [7, 17] for $M = 10, 20, 40$ and $\Delta t = 1/10^6$ at $t_f = 0.1$.

and also when the element number is increased, the error norm L_2 decreases. Again, from the table it is obvious that the newly obtained results are much more better than those in Refs. [8, 17].

Table 4.3 shows a clear comparison of the L_2 error norms computed from the new scheme using QHCM-L with those in Refs. [7, 17] for values of $\Delta x = 1/10, 1/20, 1/40$ and $\Delta t = 1/10^6$ at $t_f = 0.1$. One can obviously seen that the scheme produces good enough results and clearly, the obtained L_2 error norms are relatively smaller than those in Refs. [7, 17].

	L_2			L_∞		
$\Delta x = \Delta t$	QHCM-L	QHCM-C	[17]CHCM-L	[11]	[6](CN)	[6](CBVM)
0.2	5.1455×10^{-5}	5.1439×10^{-5}	5.8498×10^{-5}	1.4145×10^{-1}	1.1×10^{-1}	2.8×10^{-2}
0.1	3.1581×10^{-5}	3.1589×10^{-5}	3.1584×10^{-5}	3.7195×10^{-2}	3.0×10^{-2}	3.8×10^{-3}
0.05	9.7102×10^{-6}	9.7106×10^{-6}	9.7065×10^{-6}	8.4588×10^{-3}	6.9×10^{-3}	2.7×10^{-4}
0.025	2.5488×10^{-6}	2.5489×10^{-6}	2.5485×10^{-6}	2.0698×10^{-3}	1.7×10^{-3}	1.3×10^{-5}
0.0125	6.4490×10^{-7}	6.4490×10^{-7}	6.4488×10^{-7}	5.1473×10^{-4}	4.2×10^{-4}	5.1×10^{-7}
0.00625	1.6171×10^{-7}	1.6171×10^{-7}	1.6171×10^{-7}		1.1×10^{-4}	3.6×10^{-8}
0.01	4.1332×10^{-7}	4.1333×10^{-7}	4.1332×10^{-7}			
0.005	1.0353×10^{-7}	1.0353×10^{-7}	1.0353×10^{-7}			
0.0025	2.5895×10^{-8}	2.5895×10^{-8}	2.5895×10^{-8}			
0.002	1.6574×10^{-8}	1.6574×10^{-8}	1.6574×10^{-8}			
0.001	4.1437×10^{-9}	4.1437×10^{-9}	4.1437×10^{-9}			

Table 4.4: A comparison of the calculated error norms L_∞ of the present scheme with those in Refs. [6, 11, 17], for various values of $\Delta x = \Delta t$ at $t_f = 1$.

Table 4.4 displays a clear comparison of the L_∞ error norms obtained from the presented scheme with those given in Refs. [6, 11, 17] for some decreasing values of $\Delta x = \Delta t$ at $t_f = 1$. The table indicates that the computed L_∞ error norms are sufficiently small and in good harmony with those in Refs. [6, 17], but much more better than those in Ref. [11].

Finally, Table 4.5 compares the discrete values of both L_2 and L_∞ error norms for $M = 16$ and $\Delta t = 0.01$ at various t_f with those in Refs. [15, 17]. The table shows that the newly computed error norms are again small enough and better than both of those in Refs. [15, 17].

	L_2			L_∞		
t_f	QHCM-L	[17]CHCM-L	[15]	QHCM-L	[17]CHCM-L	[15]
0.1	4.2273×10^{-4}	2.9917×10^{-4}	4.86×10^{-3}	2.9889×10^{-4}	2.9891×10^{-4}	6.79×10^{-3}
0.3	1.7602×10^{-4}	1.2457×10^{-4}	8.87×10^{-5}	1.2446×10^{-4}	1.2447×10^{-4}	3.76×10^{-4}
0.5	4.0720×10^{-5}	2.8818×10^{-5}	1.73×10^{-3}	2.8791×10^{-5}	2.8793×10^{-5}	2.44×10^{-4}
0.7	7.9127×10^{-6}	5.5999×10^{-6}	2.04×10^{-4}	5.5446×10^{-6}	5.5953×10^{-6}	3.17×10^{-4}
0.9	1.4121×10^{-6}	9.9934×10^{-7}	2.14×10^{-3}	9.9840×10^{-7}	9.9856×10^{-7}	3.14×10^{-3}
1.0	5.8454×10^{-7}	4.1368×10^{-7}	2.15×10^{-3}	4.1329×10^{-7}	4.1338×10^{-7}	3.32×10^{-3}

Table 4.5: A comparison of the calculated error norms L_2 and L_∞ of the present scheme with those in Ref. [15, 17], for $M = 16$, $\Delta t = 0.01$ at various t_f.

From the presented tables one can clearly see that the newly calculated numerical results are in good harmony with both the analytical and other previously published numerical results. In fact, the present scheme in general produces better results with a smaller number of elements compared to other studies based on classical and trigonometric B-splines.
From the results given in the above tables, it is clearly seen that the approximate numerical solutions are getting closer and closer to exact ones as the mesh step sizes are refined. It should not be forgotten that the mesh step sizes must be chosen modestly small enough to avoid increase in the CPU simulation time and storage requirement capacity.

5. Conclusion

In this work, the developed fully discretized numerical scheme constructed using the conventional Crank-Nicolson-like finite difference technique for the integration of temporal quantities and a quintic Hermite B-spline finite element collocation technique for the integration of spatial quantities has been successfully applied to obtain approximate numerical solutions of the 1-D heat conduction equation for certain initial and boundary conditions. In order to confirm the accuracy and reliability of the suggested scheme, the error norms L_2 and L_∞ calculated from the exact and numerical solutions are compared with the results given by the previous ones existing in the literature. It is seen that the results obtained by applying the developed scheme to the example problem are mostly better than the previous results and at least in some cases in good agreement with the previous ones. Generally speaking, it is found a good quality harmony between the existing and the above aforesaid results. For the stability test of the current scheme, von-Neumann (Fourier) technique is performed which demonstrates that the scheme is unconditionally stable. The main contribution of this paper is the ability of the newly presented scheme to produce much better results by using less computer storage capacity and requiring less CPU time. In conclusion, the suggested scheme is producing stable, efficient and accurate results, and can be successfully used to find approximate numerical solutions of initial and boundary value problems consisting of many linear and especially nonlinear PDEs which have an important role in describing natural phenomena encountered in most branches of applied and engineering sciences.

Article Information

Acknowledgements: The authors also declare that this manuscript has been presented in The 7th International Conference on Computational Mathematics and Engineering Sciences / 20-21 May 2023, Elazığ, Türkiye.

Author's contributions: All authors contributed equally to the writing of this paper. All authors read and approved the final manuscript.

Conflict of interest disclosure: No potential conflict of interest was declared by the authors.

Supporting/Supporting organizations: No grants were received from any public, private or non-profit organizations for this research.

Ethical approval and participant consent: It is declared that during the preparation process of this study, scientific and ethical principles were followed and all the studies benefited from are stated in the bibliography.

Plagiarism statement: This article was scanned by the plagiarism program.

References

[1] A. D. Amin, E. Amin, S. Hasan, A. Meysam, L. Yueming, W. Lian-Ping, J. Dengwei, X. Gongnan, *A comprehensive review on multi-dimensional heat conduction of multi-layer and composite structures: analytical solutions*, J. Therm. Sci., **30**(6) (2021), 1875-1907.
[2] H. J. Xu, Z. B. Xing, F. Q. Wangb, Z. M. Cheng, *Review on heat conduction, heat convection, thermal radiation and phase change heat transfer of nanofluids in porous media: fundamentals and applications*, Chem. Eng. Sci., **195** (2019), 462–483,
[3] F. Suarez-Carreno, L. Rosales-Romero, *Convergence and stability of explicit and implicit schemes in the simulation of the heat equation*, Appl.Sci. **11** (2021).
[4] N. F. Kaskar, *Modified implicit method for solving one dimensional heat equation*, Int. J. Eng. Res. Comp. Sci. Eng., **8**(9) (2021), 1-6.
[5] G. Lozande-Cruz, C.E. Rubio-Mercedes, J. Rodrigues-Riberio, *Numerical solution of heat equation with singular robin boundary condition*, Tendencias em Matematica Aplicada e Computacional, **19**(2) (2018), 209-220.
[6] H. Sun, J. Zhang, *A high-order compact boundary value method for solving one-dimensional heat equations*, Numer. Methods Partial Differential Equations, **19**(6) (2003), 846-857.
[7] A. Yosaf, S. U. Rehman, F. Ahmad, M. Z. Ullah, A. S. Alshomrani, *Eight-order compact finite difference scheme for 1D heat conduction equation*, Adv. Numer. Anal., (2016), Article ID 8376061, 12 pages.
[8] F. Han, W. Dai, *New higher-order compact finite difference schemes for 1D heat conduction equations*, Appl. Math. Modell., **37** (2013), 7940-7952.
[9] S. Dhawan, S. Kumar, *A numerical solution of one dimensional heat equation using cubic b-spline basis functions*, Int. J. Res. Rev. App. Sci., **1** (2009), 71-77.

[10] H. Çağlar, M. Özer, N. Çağlar, *The numerical solution of the one-dimensional heat equation by using third degree b-spline functions*, Chaos Solitons Fractals, **38** (2008), 1197–1201.
[11] J. Goh, A. A. Majid, A. I. Ismail, *Cubic b-spline collocation method for one-dimensional heat and advection-diffusion equations*, J. Appl. Math., (2012), Article ID 458701, 8 pages.
[12] M. H. Khabir, R. A. Farah, *Cubic b-spline collocation method for one-dimensional heat equation*, Pure Appl. Math. J., **6**(1) (2017), 51-58.
[13] B. Mebrate, *Numerical solution of a one dimensional heat equation with Dirichlet boundary conditions*, Amer. J. Appl. Math., **3**(6) (2015), 305-311.
[14] N. Patel, J. U. Pandya, *One-dimensional heat equation subject to both Neumann and Dirichlet initial boundary conditions and used a spline collocation method*, Kalpa Publ. Comput., **2** (2017), 107-112.
[15] M. R. Hooshmandasl, M. H. Heydari, F. M. Maalek Ghaini, *Numerical solution of the one-dimensional heat equation by using Chebyshev wavelets method*, J. Appl. Comp. Math., **1**(6) (2012), 1-7.
[16] T. Tarmizi, E. Safitri, S. Munzir, M. Ramli, On the numerical solutions of a one-dimensional heat equation: spectral and crank Nicolson method, AIP Conference Proceedings, (2020).
[17] S. Kutluay, N. M. Yağmurlu, A. S. Karakaş, *An effective numerical approach based on cubic hermite b-spline collocation method for Solving the 1D heat conduction equation*, New Trends in Math. Sci., **10**(4) (2022), 20-31.
[18] H. Zeybek, S. B. G. Karakoç , *A collocation algorithm based on quintic b-splines for the solitary wave simulation of the GRLW equation*, Sci. Iran., **26**(6) (2019), 3356-3368.
[19] W. Wu, J. Manafian, K. A. Khalid, S. B. G. Karakoç, A. H.Taqi, M. A. Mahmoud, *Numerical and analytical results of the 1D BBM equation and 2D coupled BBM-system by finite element method*, Int. J. Mod. Phys. B, **36**(28) (2022).
[20] S. B. G. Karakoç, Y. Uçar, N. M. Yağmurlu, *Different linearization techniques for the numerical solution of the MEW equation*, Selcuk J. Appl. Math., **13**(2) (2012), 43-62.
[21] S. Arora, I. Kaur, W. Tilahun, *An exploration of quintic Hermite splines to solve burgers' equation*, Arab. J. Math., (2020), 19–36.
[22] S. Arora, I. Kaur, *Applications of quintic Hermite collocation with time discretization to singularly perturbed problems*, Appl. Math. Comput., **316** (2018) 409–421.
[23] S. P. Kaur, A. K. Mittal, V. K. Kukreja, A. Kaundal, N. Parumasur, P. Singh, *Analysis of a linear and non-linearmodel for diffusion–dispersion phenomena of pulp washing by using quintic hermite interpolation polynomials*, Afr. Mat., **32** (2021), 997–1019.

Journal of Mathematical Sciences and Modelling, 7(2) (2024) 90-111
Research Article

Journal of Mathematical Sciences and Modelling

Journal Homepage: www.dergipark.gov.tr/jmsm
ISSN: 2636-8692
DOI: https://doi.org/10.33187/jmsm.1505481

Modeling and Analysis of Kamikaze *UAV* Design with 3 Different Wing Configurations

Ahmed Receb Demirel[1], Mustafa Murat Yavuz[2] and Nehir Tokgöz[1*]

[1] *Faculty of Engineering, Sakarya University, Sakarya, Türkiye*
[2] *Faculty of Engineering, Izmir Democracy University, Izmir, Türkiye*
[*] *Corresponding author*

Article Info

Keywords: *Angle of Attack, Kamikaze, UAV, Pressure*
2010 AMS: *39A13, 39A28, 39A30, 92D30*
Received: *26 June 2024*
Accepted: *15 August 2024*
Available online: *28 August 2024*

Abstract

Appropriate design parameters need to be determined for unmanned aerial vehicles that can perform kamikaze missions. In this study, a *UAV* with 3 different wing configurations and a fuselage and tail wings were designed, and the flow around the wing was examined using computational fluid mechanics. Advanced modeling techniques were employed to simulate and analyze the aerodynamic behavior of these configurations. The effect of angle of attack (AoA), wing positioning on the fuselage, and wing configurations were investigated. Due to the effect of the wing sweep angle, high-pressure values in the arrow-angle wing were lower than in rectangular and trapezoidal wings. In a similar situation, the flow separation on the arrow-angle wing is less advanced towards the wing tip. When the wing type and connection location were examined, the highest C_l/C_d ratio was obtained in the trapezoidal model connected to the fuselage in the middle. The results of numerical wing models compared with the theoretical lift coefficient were consistent. Trapezoidal and rectangular wings had a high lift coefficient, but after 15° of AoA, the lift coefficient decreased. At angles of attack beyond 15°, the arrow-angle wing still has an increasing lift coefficient. As the angle of attack increased, the drag coefficient was also enhanced. The lowest drag coefficient occurred in the arrow-angle wing model. Up to 5°of AoA, all wing models raised the C_l/C_d ratio. The C_l/C_d ratio decreased at higher angles of attack. As a result of the examination, it would be more correct to choose trapezoidal and arrow-angle wings.

1. Introduction

Unmanned aerial vehicles (*UAVs*), often colloquially termed drones, represent a class of aircraft capable of autonomous flight without human intervention. The evolution of *UAVs* has been spurred by the progressive complexity and advancements within the domains of control systems, robotics, electronics, and artificial intelligence [1]. These technological strides have profoundly influenced the design, production, and versatility of drones, broadening their applications across numerous industries and sectors. *UAVs* are now extensively employed in logistics [2, 3], agriculture (including crop protection) [4], construction site monitoring, cinematography [5, 6], surveillance operations [7], smart city initiatives [8], disaster response efforts [9], healthcare logistics, online shopping transportation [10], remote sensing missions [11], and provision of wireless connectivity [12]. This multifaceted utility underscores the pivotal role that *UAVs* play in modern society, revolutionizing various fields through their unparalleled aerial capabilities. The versatile applications of drones, a technology drawing significant interest and continuously progressing, have emerged as unexpectedly powerful tools, even in the Ukrainian conflict [13]. Notably, *UAVs* are employed for tasks such as crowd monitoring, dissemination of public information, and sanitization efforts aimed at combating the COVID-19 pandemic [14]. Furthermore, efforts are underway to develop methods for utilizing authorized *UAVs* in tracking and surveilling potentially threatening unmanned aerial vehicles, demonstrating the ongoing innovation and adaptability within the realm of drone technology [15]. The adoption of drones has revolutionized the execution of monotonous, hazardous and labor-intensive tasks, offering enhanced efficiency and reduced expenses while mitigating risks to human safety [16]. By substituting traditional methods with drones, substantial cost savings can be achieved alongside widespread acceptance, while also augmenting the value of collected data [17].

***Email addresses and ORCID numbers*: ahmedrecebd@gmail.com, 0009-0002-5120-3353 (A. R. Demirel), murat.yavuz@idu.edu.tr, 0000-0002-5892-0075 (M. M. Yavuz), nehirtokgoz@sakarya.edu.tr, 0000-0001-9264-9971 (N. Tokgöz)**
Cite as "A. R. Demirel, M. M. Yavuz, N. Tokgöz, *Modeling and analysis of Kamikaze UAV design with 3 different wing configurations*, J. Math. Sci. Model., 7(2) (2024), 90-111"

With their affordability and heightened maneuverability, adaptability, safety features, and customization options, drones have become indispensable tools [18, 19]. Recent studies indicate a remarkable growth trajectory in the global drone market, with projections soaring to $19.85 billion USD by 2021, marking a compound annual growth rate of nearly 13% over the forecast period [20]. Nonetheless, it is essential to acknowledge the mounting concerns regarding cybersecurity associated with drone applications. A significant constraint for drones revolves around their range and autonomy, which are influenced by factors such as battery capacity and aerodynamic performance [20]. Over recent years, various approaches have been explored to enhance drone efficiency, with particular focus on methods like implementing winglets [21]. Prieto et al. [13] conducted an aerodynamic enhancement of a "Vertical Take Off and Landing" drone by integrating winglets and wingtips. They evaluated two approaches for this analysis: the Vortex Lattice Method and Computational Fluid Dynamics modeling. By comparing the results from OpenFOAM and XFLR5, it was observed that, XFLR5 tends to overestimate lift and viscous drag while underestimating induced and total drag. Quintana et. al [22], introduced and examined an optimized morphing unmanned aerial vehicle (UAV) designed to dynamically adjust, and extend its wings for enhanced aerodynamic and structural performance. Their study aimed to assess and enhance the *UAV's* efficiency, analyze flight transition dynamics, and ensure structural integrity. The modal analysis results indicated that, the wing's low natural frequency increases the likelihood of flutter and vibratory response induced by aerodynamic forces and disturbances. Voskuijl [23] conducted a study to offer an in-depth technical analysis of the current stray munitions, focusing on their design and the performance of the aircraft that deploy them. To achieve this goal, the scientist created a detailed database of stray munitions using information sourced from publicly available data. Six primary configurations were identified: (1) conventional, (2) delta wing, (3) tandem wing, (4) canard, (5) cruciform and (6) rotorcraft. According to the results, the cruciform configuration was beneficial for precise flight path control. The tandem wing configuration integrated the advantages of a canister launch with high aspect ratio wings, making it suitable for long-range flights, while the delta wing design provided a large internal volume and reaches high terminal attack airspeed. Zampronha et al. [24] examined the Iranian HESA Shahed 136 drone to explore recent advancements in low-cost, long-range precision weapons, with a focus on kamikaze drones and stray munitions. The HESA Shahed 136 has transformed the approach to precise long-range strikes a capability previously exclusive to costly and technologically sophisticated tactical missiles and aircraft. Now, this function can be performed with inexpensive drones, igniting arms race not only to develop the most advanced and precise weaponry, but also to create the most affordable solution. Experimental analysis was carried out by Saraçyakupoğlu et al. [25] within the framework of four different scenarios to calculate the *UAV* and determine the design criteria. The *UAV* under discussion was designed with extensive operational capability, particularly suited for defense and border security tasks. The research suggested that the *UAV's* mid-wing, double-tail design, along with its relatively light body, enhanced its three-axis stability, providing notable benefits, particularly concerning operational cost. In the study conducted by Sakarya et al. [26], the aim was to develop an unmanned helicopter project that could be used in the military field. The helicopter they developed, which can be controlled autonomously or by a pilot, was designed for areas, where soldier visibility or access is difficult and dangerous. It is a prototype product intended for aerial reconnaissance, surveillance, and, in certain situations, it can perform a suicide dive thanks to the explosive it carries.

The literature review clearly indicates a growing momentum in the research and development of drones and model airplane-sized unmanned aerial vehicles (*UAVs*). These studies have predominantly focused on advancing the design and functionality of *UAVs*, driven by the increasing demand for innovative applications across various industries. In this study, we have sought to contribute to this evolving field by developing a novel unmanned aerial vehicle with a distinctive usage area, employing advanced numerical methods. A comprehensive analysis was conducted to investigate the impact of the wing structure and the positioning of the wing/fuselage joint on the *UAV's* performance. This approach not only aims to enhance the aerodynamic efficiency of the *UAV* but also to explore new design possibilities that could expand its operational capabilities.

2. Material and Method

2.1. Design parameters and models

The design consists of the fuselage, fuselage wing, and tail wing. In general, a parametric formula regarding body length was used. In the parametric formula taken from the literature, homemade composites were chosen due to their lightness and ease of production, and alpha and C coefficients were used as 3.50 and 0.23 [27-28]. The maximum aircraft length value, generally 6 kg, is calculated in Eq. (2.1) and Eq. (2.2) below. According to the result obtained from Eq. (2.2), the body length should not exceed 1.932 $meters$. The hull length for the design was determined as 1.850 $meters$. The width of the body is 0.210 $meters$ and the height is 0.205 $meters$. The body design of the kamikaze *UAV* model is shown in Figure 2.1.

$$W_0 = 6kg = 13.23Ib \tag{2.1}$$

$$length\,(feet) = a \times W_0^c = 3.5 \times 13.23^{0,23} = 6.339feet = 1.932m \tag{2.2}$$

According to the result obtained from Eq. (2.2), the body length should not exceed 1.932 $meters$. The hull length for the design was determined as 1.850 $meters$. The width of the body is 0.210 $meters$, and the height is 0.205 $meters$. The body design of the kamikaze *UAV* model is shown in Figure 2.1

After determining the fuselage dimensions, the wingspan ratio (AR) must first be determined for wingspan sizing. This ratio varies depending on the aircraft type. C_d is a dimensionless parameter used to measure the drag of a wing. The C_d coefficient is a function of the angle of attack (AoA). As the angle of attack increases, the C_d coefficient increases and therefore more thrust is required. Drag coefficient; C_d [2];

Figure 2.1: Kamikaze *UAV* body design

$$C_d = \frac{(2 \times D)}{(\rho \times V^2 \times S)} \tag{2.3}$$

In this equation, D: drag force (N), ρ: density of air $\left(1.225\ kg/m^3\right)$, V: speed (m/s), S: wing area m^2. Wing taper can change wing lift distribution. This is assumed to be an advantage of the taper, as it is a technical means to improve lift distribution. The aspect ratio of the wing was used as a span ratio of 7.27, within the range of 5-9 ratio used in general aviation structure. Technical drawings and basic dimensions for arrow-angle, trapezoidal and rectangular wings were shown in Figure 2.2. Basic geometrical properties of the wings were given in the Table 2.1.

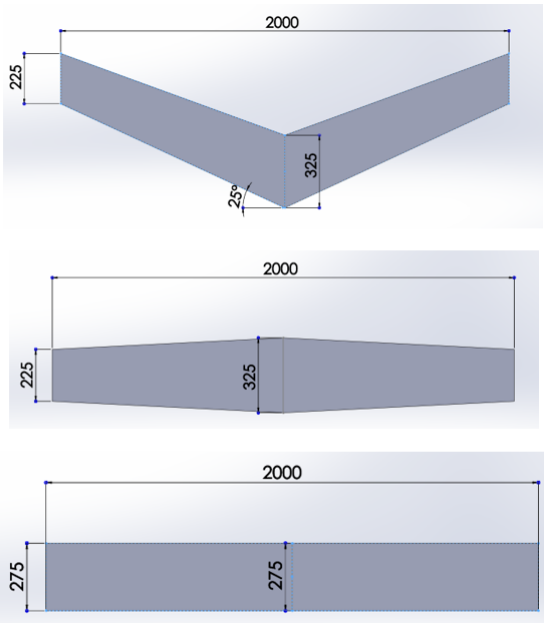


Figure 2.2: Arrow-angle, trapezoidal and rectangular Wings (all lengths (mm))

NACA 4412 profile was preferred on the wings due to its ability to provide high lift force at small angles of attack at relatively low Reynolds numbers, and NACA 0012 profile, which does not contain an asymmetrical situation due to stabilization control in the tail, was preferred. V-tail geometry was used in the tail and a rear-wheel drive propeller was preferred. Reynolds number is the ratio of inertial forces to viscosity forces of a fluid, and this value gives the relative importance of these two types of forces to each other under a certain flow condition.

Table 2.1: Wing geometry dimensions

Wing	Arrow-angle (mm)	Trapezoidal (mm)	Rectangular (mm)
$c_{root}\ (Root\ chord\ length)$	325	325	275
$c_{tip}\ (Tip\ chord\ length)$	225	225	275
$c_{ave}\ (Ave.chord\ length) = \frac{c_{root}+c_{tip}}{2}$	275	275	275
$b\ (Wing\ span)$	2000	2000	2000
$AR = b/c_{ort}$	7.27	7.27	7.27
$S\ (Wing\ area) = b \times c_{ort}$	0.55	0.55	0.55
$\lambda\ (Tapering\ rate) = c_{tip}/c_{root}$	0.6923	0.6923	1
λ	1	1	1
$c_{mgc}\ (Average\ aerodynamic\ chord\ length) = (2/3) \times c_{root} \times \left(1+\lambda+\lambda^2/(1+\lambda)\right)$	170.51	170.51	275

Therefore, the Reynolds number is frequently used to characterize different flow regimes, such as uniform flow and turbulent flow. The given formula (2.4) is used for calculation of Reynolds number;

$$Re = \frac{\rho \times V \times L}{\mu} = \frac{V \times L}{v} \tag{2.4}$$

$\rho =$ density $\left(kg/m^3\right)$

$V =$ flow velocity $\left(m/s^2\right)$

$L =$ the length of the pipe through which the fluid flows (m)

$\mu =$ dynamic viscosity of the fluid $(Pa.s)$

$v =$ kinematic viscosity of the fluid $\left(m^2/s\right)$

Based on the equation, it can be decided whether the flow is laminar or turbulent. Reynolds number value ranges for external flow (Eq. (2.5) and (2.6)) ;

$$\text{laminer}: Re \leq 5 \times 10^5 \text{ , turbulent} : 5 \times 10^5 \leq Re \leq 10^7 \tag{2.5}$$

$$Re = \frac{1.225 \times 25 \times 0.275}{1.789 \times 10^{-5}} = 470758.804 \tag{2.6}$$

According to the calculated Reynolds number, the flow is turbulent. The boundary layer thickness is calculated in Eq. (2.7) and Eq. (2.8) Turbulence intensity is given in Eq's. (2.9)-(2.11). This turbulence intensity was used in the free entry of the flow;

$$\delta_L = \frac{0.37 \times c}{Re^{\frac{1}{5}}} \text{ (Total boundary layer thickness)} \tag{2.7}$$

$$\delta_L = \frac{0.37 \times 0.275}{470758.804^{\frac{1}{5}}} = 7.464 \times 10^{-3} \tag{2.8}$$

$I =$ turbulence intensity;

$$Re_g = \frac{\rho \times V \times \Delta h}{\mu} = \frac{1.225 \times 25 \times 1.5}{1.789 \times 10^{-5}} = 2567775.293 \tag{2.9}$$

$$I = 0.16 \times Re_g^{-\frac{1}{8}} = 0.16 \times 2567775.293^{-\frac{1}{8}} = 0.02529 \tag{2.10}$$

$$I = 2.529\% \tag{2.11}$$

The flow field is considered viscous, turbulent, and incompressible to exclude the solution of energy equations. The applicable equations for the current RANS (Reynolds-Averaged Navier-Stokes) calculations consist of the continuity equation and the momentum conservation equation, which are expressed as follows in Eq. (2.12) and Eq. (2.13): [27]
Continuity Equation;

$$\frac{\partial \rho}{\partial t} + \nabla (\rho \overrightarrow{u}) = 0 \tag{2.12}$$

Momentum Equation;

$$\frac{\partial (\rho \overrightarrow{u})}{\partial t} + \nabla (\rho \overrightarrow{u} \overrightarrow{u}) = \nabla (\mu \nabla \overrightarrow{u}) - \nabla p + \rho \overrightarrow{g} \tag{2.13}$$

Here, $\overrightarrow{u}$ denotes the velocity vector in the x, y and z directions, p denotes the pressure, μ denotes the dynamic viscosity, and ρ denotes the fluid density.
It was decided to use the $k-\omega$ sst turbulence model equation (2.4) (2.5) due to its ability to accurately model adhesion loss and flow separation of airfoils among the results stated in the literature [28]. The grid structure of the model to be used in flow analysis is important. In 2D wing examinations, the total thickness parameter on the wing was modelled as a grid with 25 layers based on the result shown by the mesh (grid) program (first model) and the results were obtained. Lift and drag coefficients are shown in Table 2.2. After obtaining the results with the first model, the results were obtained by creating layers (final model) with the first layer by taking $y = +1$ and increasing by 35 layers. The latest model structure was used in the investigations.

$$\frac{\partial (\rho k)}{\partial t} + div(\rho k U_i) = div\left[\left(\mu + \frac{pk}{\omega \sigma_k}\right) \nabla k\right] + 2\frac{(\rho k)}{\omega} S_{ij}.S_{ij} - \frac{2}{3}\rho k \frac{\partial U_i}{\partial x_j} S_{ij} - \beta * \rho k w$$

$$\frac{\partial (\rho \omega)}{\partial t} + div(\rho \omega U_i) = div\left[\left(\mu + \frac{\rho k}{\omega \sigma_{\omega^1}}\right) \nabla \omega\right] + 2\rho \gamma_2 S_{ij}.S_{ij} - \frac{2}{3} p \gamma_2 \omega \frac{\partial U_i}{\partial x_j} S_{ij} - \beta_2 \rho \omega^2 + 2\frac{\rho}{\omega \sigma_{\omega^2}} \frac{\partial k}{\partial x_k} \frac{\partial \omega}{\partial k_k}$$

Table 2.2: C_l and C_d values at different angles of attack in 2D analysis

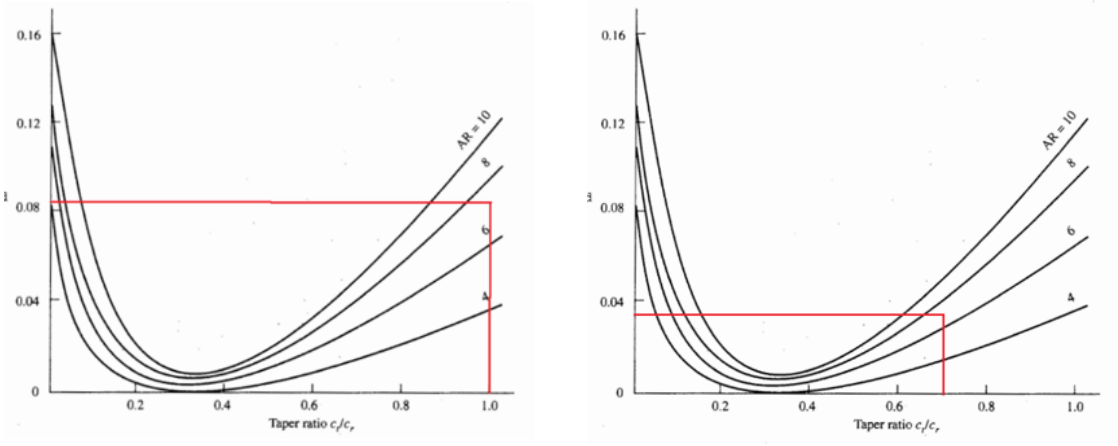
	First model (initial)			Last model (stable)		
AoA	C_l	C_d	C_l/C_d	C_l	C_d	C_l/C_d
0	0.449	0.0227	19.756	0.466	0.0174	26.698
2	0.670	0.1819	36.842	0.695	0.0193	36.024
5	0.960	0.0227	42.227	1.004	0.0239	31.861
10	1.331	0.0378	35.148	1.381	0.0390	35.350
12	1.412	0.0488	28.920	1.462	0.0500	29.233
13	1.426	0.0560	25.440	1.495	0.0579	26.826
14	1.419	0.0658	21.560	1.504	0.0681	22.078
15	1.395	0.0808	17.262	1.483	0.0823	18.019

2.2. Model verification

The effect of the number of elements was examined for 3D numerical models used in examining wing structures. Drag and lift coefficients, which are a result of aerodynamic effects, were used as the basis for validation. 3 different wing designs were investigated. In Table 2.3, C_l and C_d results of numerical analyses were given.

Table 2.3: C_l and C_d values in different wings according to the number of elements

Wing type	Number of elements	C_l	C_d	C_l/C_d
rectangular	5007723	0.3259	0.0208	15.667
	2286244	0.3496	0.2126	16.443
	1307221	0.3605	0.0214	16.773
	870499	0.3656	0.0217	16.842
trapezoidal	5074373	0.3308	0.02064	16.025
arrow-angle	9361220	0.2975	0.01976	15.055
	6241939	0.31405	0.0201158	15.612
	5178034	0.31462	0.0201492	15.612
	5186810	0.32481	0.0207111	15.683
	2348390	0.33392	0.0204356	16.340
	1341935	0.34584	0.0206863	16.718
	965296	0.35246	0.0209365	16.835



(a) **(b)**

Figure 2.3: Induced drag factor, δ(a: rectangular; b: trapezoidal)

2.2.1. Rectangular wing lift coefficient verification

At the outset of the 3D model analysis, the quantity of elements was analyzed for considering the lift coefficient (C_l) and drag coefficient (C_d) of the rectangular wing. The findings are summarized in Table 3. It was determined that stable results were achieved using the highest element number from the table, and subsequent examinations were conducted at this element density. The maximum value was then considered the optimal mesh and compared to the theoretical lift coefficient (C_l) value. Induced drag factor was taken from Figure 2.3a for a taper ratio of 1 and AR was 7.27; $\delta = 0.083$.

$$
\begin{aligned}
a &= \frac{dC_L}{d\alpha} \\
a &= \frac{a_0}{1+\frac{a_0}{\pi \times e \times AR}} \\
e &= \frac{1}{1+\delta} = \frac{1}{1+0.083} = 0.92336 \\
\delta &\cong 0.083 \text{ (Induced drag factor)} \\
a &= \frac{6.02}{1+\frac{6.02}{\pi \times 0.92336 \times 7.27}} = 4.68315 \text{ radian} \\
a &= \frac{4.68915}{57.3} = 0.08173 \text{ deg} \\
C_L &= a \times (\alpha - \alpha_{L=0}) = 0.08173 \times (0-(-4)) = 0.326921 \\
\text{Error ratio} &= \frac{\left|C_{L\,analy.} - C_{L\,real}\right|}{C_{L\,real}} \times 100 = \frac{|0.329121 - 0.326921|}{0.326921} \times 100 = 0.67\%
\end{aligned}
$$

Here, the first α in parentheses is 0°, which is the angle of the wing. $\alpha_L = 0$ is the angle of attack at which the airfoil is at zero lift.

2.2.2. Trapezoidal wing lift coefficient verification

The error ratio was calculated for trapezoidal wing and was given in Eq. 27. δ was taken from Figure 2.3b. The taper ratio was $225\,mm/325\,mm = 0.6923$ which was shown previously in Figure 2.2. $AR = 7.27$ and δ is obtained 0.035.

$$
\begin{aligned}
a &= \frac{dC_L}{d\alpha} \\
a &= \frac{a_0}{1+\frac{a_0}{\pi \times e \times AR}} \\
\delta &\cong 0.035 \text{ (Induced drag factor)} \\
e &= \frac{1}{1+\delta} = \frac{1}{1+0.035} = 0.966183 \\
a &= \frac{6.02}{1+\frac{6.02}{\pi \times 0.966183 \times 7.27}} = 4.729711 \text{ radian} \frac{n!}{r!\,(n-r)!} \\
a &= \frac{4.729711}{57.3} = 0.0825429\ deg \\
C_L &= a \times (\alpha - \alpha_{L=0}) = 0.082635 \times (0-(-4)) = 0.330171 \\
\text{Error ratio} &= \frac{\left|C_{L\,analy.} - C_{L\,real}\right|}{C_{L\,real}} \times 100 = \frac{|0.333242 - 0.330171|}{0.330171} \times 100 = 0.93\%
\end{aligned}
$$

2.2.3. Arrow angle wing list coefficient verification

Arrow-angle wing lift coefficient was calculated for a 25° back arrow angle.

$$
\begin{aligned}
a &= \frac{dC_L}{d\alpha} \\
\cos(\wedge) &= \cos(25)\ (back\,arrow\,angle) \\
a &= \frac{a_0 \times \cos(\wedge)}{\sqrt{1+\left(\frac{a_0 \times \cos(\wedge)}{\pi \times AR}\right)^2} + \frac{a_0 \times \cos(\wedge)}{\pi \times AR}} \\
a &= \frac{6.02 \times \cos(25)}{\sqrt{1+\left(\frac{6.02 \times \cos(25)}{\pi \times 7.27}\right)^2} + \frac{6.02 \times \cos(25)}{\pi \times 7.27}} = 4.306509\ radian \\
a &= \frac{4.306509}{57.3} = 0.07515\ deg \\
C_L &= a \times (\alpha - \alpha_{L=0}) = 0.07515 \times (0-(-4)) = 0.3006 \\
\text{Error ratio} &= \frac{\left|C_{L\,analy.} - C_{L\,real}\right|}{C_{L\,real}} \times 100 = \frac{|0.297495 - 0.3006|}{0.3006} \times 100 = 1.033\%
\end{aligned}
$$

Error rates were deemed to be acceptable, and investigations continued.

3. Results and Discussion

3.1. Lift and drag coefficients on wings

Subsequent to juxtaposing the theoretical predictions with the empirical findings, varying angles of attack influenced the lift (C_l) and drag (C_d) coefficients were explored. The C_l, C_d and C_l/C_d results were given in Tables 3.1, 3.2 and 3.3 for rectangular, trapezoidal and arrow-angle wing, respectively. Upon analysis of the lift coefficient, it was observed that for all models, the lift coefficient escalated with the angle of attack (AoA) up to 15°. Notably, the trapezoidal wing exhibited the most substantial lift coefficient. In contrast, the rectangular wing demonstrated lift coefficients comparable to the trapezoidal wing, yet its lift coefficient ceased to rise beyond an AoA of 12°. Beyond an angle of attack (AoA) of 15°, both the rectangular and trapezoidal wings experienced a decline in their lift coefficients. Conversely, the arrow-angle wing model, despite having a lower lift coefficient initially, continued to see an increase in its lift coefficient until reaching an AoA of 20°.

As for the drag coefficient, the drag coefficient of all wing models increases with increasing AoA. Rectangular and trapezoidal wings gave similar results, while arrow-angle wing showed lower drag coefficient. For all models, the ratio of lift coefficient to drag coefficient (C_l/C_d) rises until reaching an angle of attack of 5°. Beyond this angle, the ratio begins to decline. The lift-to-drag ratio (C_l/C_d) for the trapezoidal wing model surpassed others until an angle of attack of 14°, whereas the arrow-angle wing model's lift-to-drag ratio peaked at angles of attack exceeding 14°.

Table 3.1: Rectangular wing cl and cd values according to angles of attack

$AoA\,(^\circ)$	C_l	C_d	C_l/C_d
0	0.32912137	0.020896613	15.74998637
5	0.73337471	0.037799812	19.40154385
10	1.0710421	0.069095768	15.50083501
12	1.1446231	0.088171176	12.98182866
14	1.1431234	0.11931991	9.580324021
15	1.1657	0.13733	8.488312823
16	1.1204377	0.15640722	7.163593215

Table 3.2: Trapezoidal wing C_l and C_d values according to angles of attack

$AoA\,(^\circ)$	C_l	C_d	C_l/C_d
0	0.32912137	0.020896613	15.74998637
5	0.73337471	0.037799812	19.40154385
10	1.0710421	0.069095768	15.50083501
12	1.1446231	0.088171176	12.98182866
14	1.1431234	0.11931991	9.580324021
15	1.1657	0.13733	8.488312823
16	1.1204377	0.15640722	7.163593215

Table 3.3: Arrow-angle wing C_l and C_d values according to angle of attack

$AoA\,(^\circ)$	C_l	C_d	C_l/C_d
0	0.29749516	0.01976	15.055423
5	0.67764535	0.03507	19.321561
10	0.999483325	0.06286	15.825955
12	1.0812807	0.07832	13.80603
14	1.1373059	0.09892	11.496681
15	1.1560672	0.11254	10.272294
17	1.1951	0.14812	8.068458
18	1.2143577	0.17092	7.1047906
19	1.2318	0.19423	6.3419657
20	1.23	0.22149	5.5532981

3.2. Pressure contours formed on wings

Imaging the flow around a wing is complex, and pressure contours are widely used to express this interaction. The effects of different angles of attack are compared for rectangular, trapezoidal and arrow-angle wing models and are shown in Figures 3.3, 3.4 and 3.5. Based on the side view of the wings, a plane was created at a distance 0.5 m from the wing-fuselage junction and the pressure contours formed in this plane are shown. Results up to 16° angle of attack on rectangular and trapezoidal wings and up to 18° angle of attack on arrow-angle wings are shown and compared with each other. For the values shown by the contours, the highest positive pressure is labelled with the letters H++ (red colour), the intermediate pressure is labelled with the letters H+ (orange) and H (yellow). In negative pressures, the highest value is labelled with the letters L– (dark blue), medium pressure L- (light blue) and low-pressure L (green). It was observed that at 0° angle of attack, there was a negative low pressure (L) on the upper surface of all wings and a low positive pressure under the wings. There was the

highest positive pressure at the front end of the wing due to the edge that interacts the flow. While the highest pressure occurred at the H++ level in the rectangular and trapezoidal wing structure, it occurred at the H+ level in the arrow-angle wing. It can be said that the sweep angle effect in the arrow-angle wing provides less pressure formation at the wing tip. It was observed that a negative low pressure (L) occurred at the bottom of the leading edge of the wings. NACA 4412 airfoil is not fully symmetrical. The fact that the underside of the wing is less humped than the top of the wing ensured the formation of this low pressure after the separation of the flow at the wing tip.

For the trapezoidal wing, the outcome at a 2° angle of attack indicated the clear formation of an L- pressure on the wing's surface. The pressure at the tip of the wing has intensified, leading to a significant increase in H++ pressure. However, this high-pressure area has become smaller and is now mostly confined to the wing's tip. When analysing the impact at a 5° angle of attack, it's observed that the pressure patterns on both rectangular and trapezoidal wings are quite alike. With an increase in the wing's angle of attack, the zone of maximum pressure, denoted as H++, has moved a bit from the center of the wing's tip towards the wing's underside. All wings have developed a negative medium pressure, labelled as L-. Nonetheless, the positive pressure beneath the wing isn't substantial. In contrast to other wing shapes, the arrow-angle wing doesn't exhibit the highest pressure, H++, at its front edge.

At 10° angle of attack, the teardrop aerofoil's lift became more pronounced. The peak positive pressure shifted from the wing's tip to its underside and the high-pressure zone, H+ enlarged. Meanwhile, the wing's negative pressure regions also increased. The arrow-angle wing did not experience the highest pressure, H++, and the negative pressure area, L–, was smaller and located in front of the wing's upper chamber. When the angle increased to 12°, the separation of pressure at the wing's front edge was noticeable, with positive pressure at H level. Both L– and H+ pressure areas got bigger. The negative pressure, L, extended over the wing's top surface to its back end. The L zone was most extensive on rectangular wings and smallest on arrow-angle wings. At 14° *AoA*, the trapezoidal wing predominantly showed high pressure underneath and low-pressure variations on top. The negative pressure region, L, expanded across all wing types. The area with the utmost positive pressure, H++, remained the same on rectangular and trapezoidal wings but reduced on the arrow-angle wing.

The airflow patterns around the wings changed noticeably at a 15° angle of attack. For rectangular and trapezoidal wings, the negative pressure zones above the wing decreased, whereas they increased for the arrow-angle wing. This trend continued at 16°. At 17 and 18°, only results for the arrow-angle wing were observed. Despite increasing angles of attack, the negative pressure on the wing developed and expanded smoothly. The positive pressure distribution underneath the wing remained unaffected.

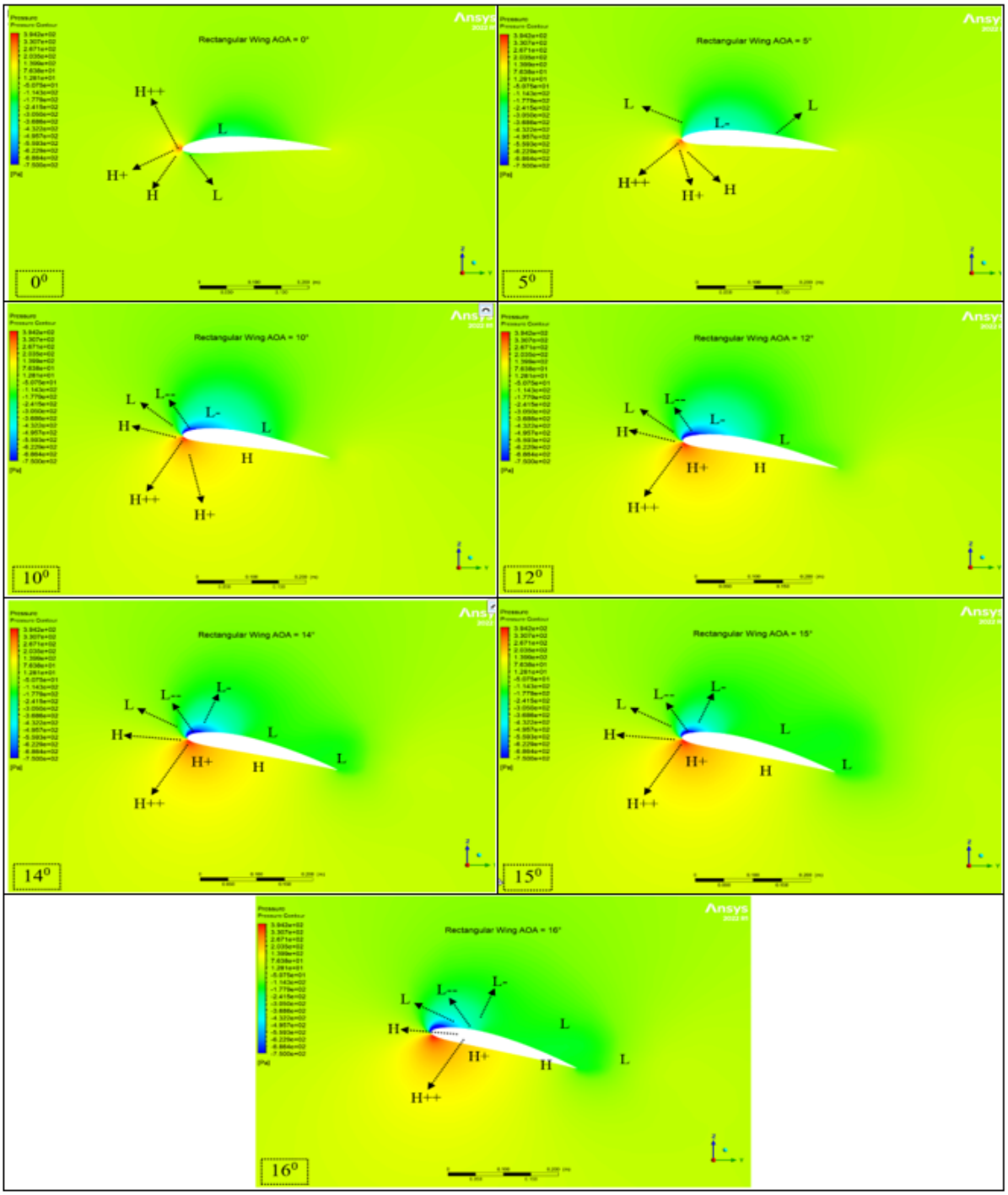


Figure 3.1: Pressure distributions of rectangular wing at $0-5-10-12-14-15-16^{\circ}$ of angles of attack

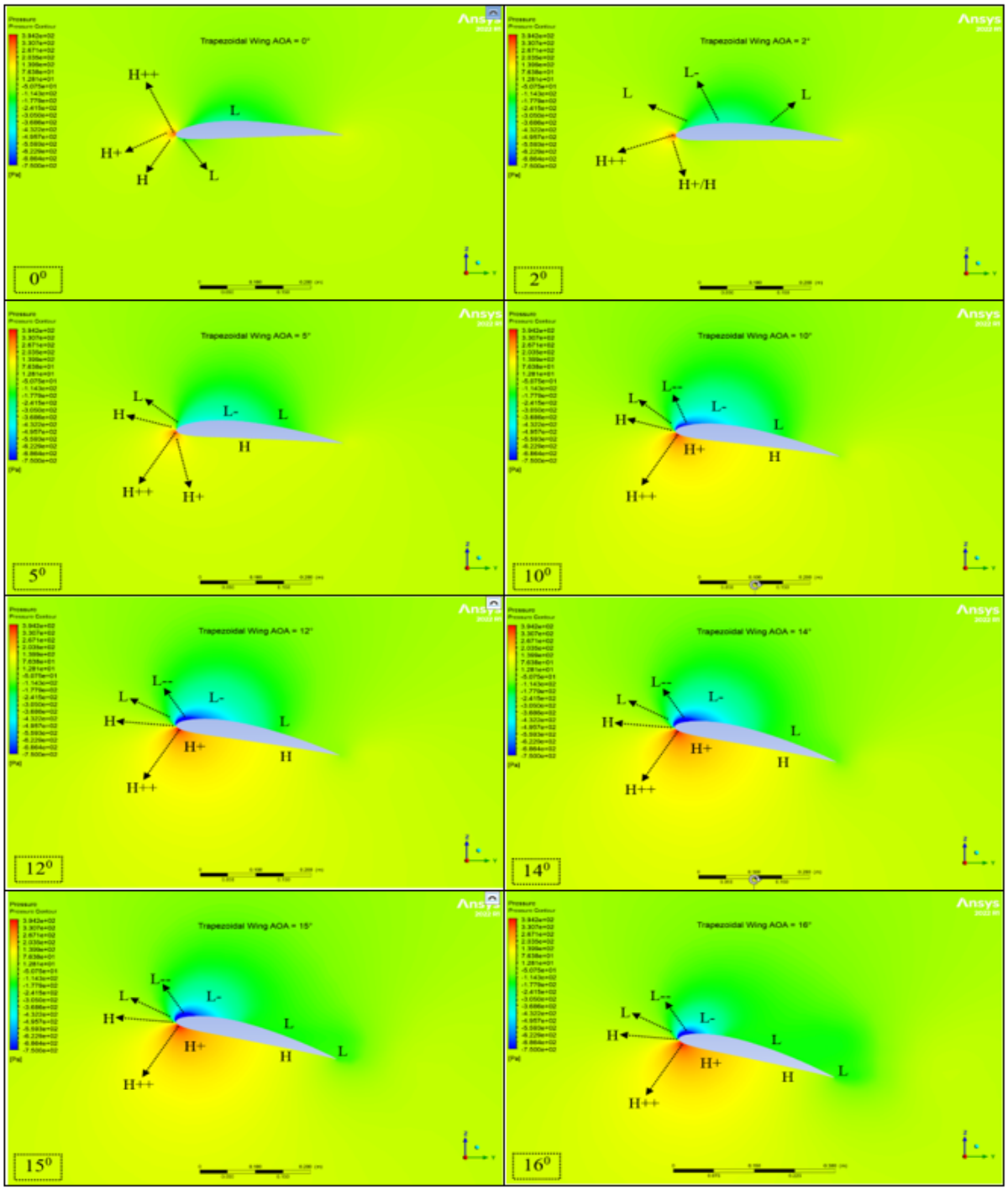


Figure 3.2: Pressure distributions of trapezoidal Wing at $0 - 2 - 5 - 10 - 12 - 14 - 15 - 16^{\circ}$ of Angles of Attack

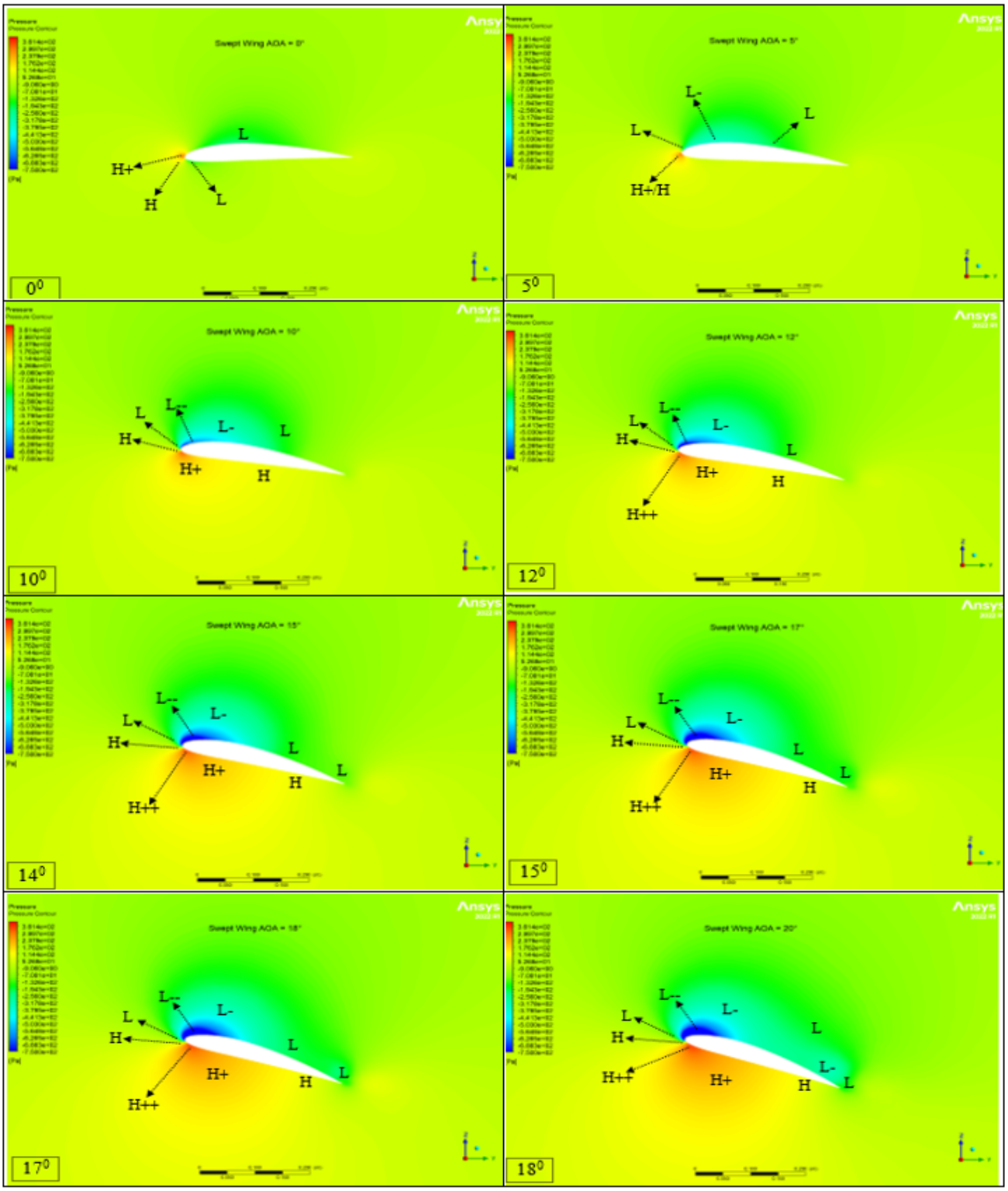


Figure 3.3: Pressure distributions of arrow angle wing at $0-5-10-12-14-15-17-18^{\circ}$ of angles of attack

3.3. Velocity contours around wings

The velocity contours formed around the wings are shown in Figures 3.4, 3.5 and 3.6 for rectangular, trapezoidal and arrow-angle wings. Based on a free flow speed of 25 m/s, regions of low-speed formation were identified near the leading edge and behind the wing. Due to the teardrop wing profile structure, the flow velocity increased on the upper surface of the wing. Furthermore, there's an observable increase in flow velocity within a specific area along the bottom of the wing's leading edge. The formation of low pressure with increasing velocity becomes more evident when comparing speed and pressure contours simultaneously. At tha 2° angle of attack, the high-speed region beneath the trapezoidal wing decreased. Increasing the angle of attack to 5° led to an expansion of the high-speed area on the wing, particularly noticeable in the trapezoidal and arrow-angle wing designs. As the flow speed increases over the upper surface of the trapezoidal wing, separation of flow from the wing surface begins to occur at the rear of the wing camber.
At a 10° angle of attack, an SP was noted on all three types of wings. With an increase in the angle of attack, the low-velocity profile, which appears as a wake trailing the wing, extended from the rear of the wing to its upper side. In the arrow-angle wing, the low velocity profile on the wing covers a smaller area. When the angle of attack is set to 12°, the wing tips experience the highest velocity, while simultaneously, a region of lower velocity is created just beneath the wing tips because of the wings separating the airflow. At 14° angle of attack, the velocity variations across different wing shapes are noticeable. For the rectangular wing, the stagnation point is at the wing's camber peak, creating a broad zone of reduced the velocity behind it. The trapezoidal wing has a smaller area of low velocity on its surface. Yet, the most pronounced effect is the increased velocity at the wing's leading edge. With the arrow-angle wing, the stagnation point is found towards the back, resulting in a sparse area of low velocity. The extensive low-velocity profile generated by a 15° rectangular wing behind it covers a large area, which is not conducive for optimal wing performance. In the trapezoidal wing, the low-velocity region has expanded, and the separation point has shifted towards the wing's leading edge. However, there was no alteration observed in the profile of the arrow-angle wing. At a 16° angle of attack, overwing flow disruption occurs in the trapezoidal wing. For arrow-angle wings at attack angles of 17° and 18°, it's observed that there's a significant area of low velocity beneath the wing. At an 18° angle of attack, the airflow is noted to create a reverse flow pattern in the low velocity region above the wing.

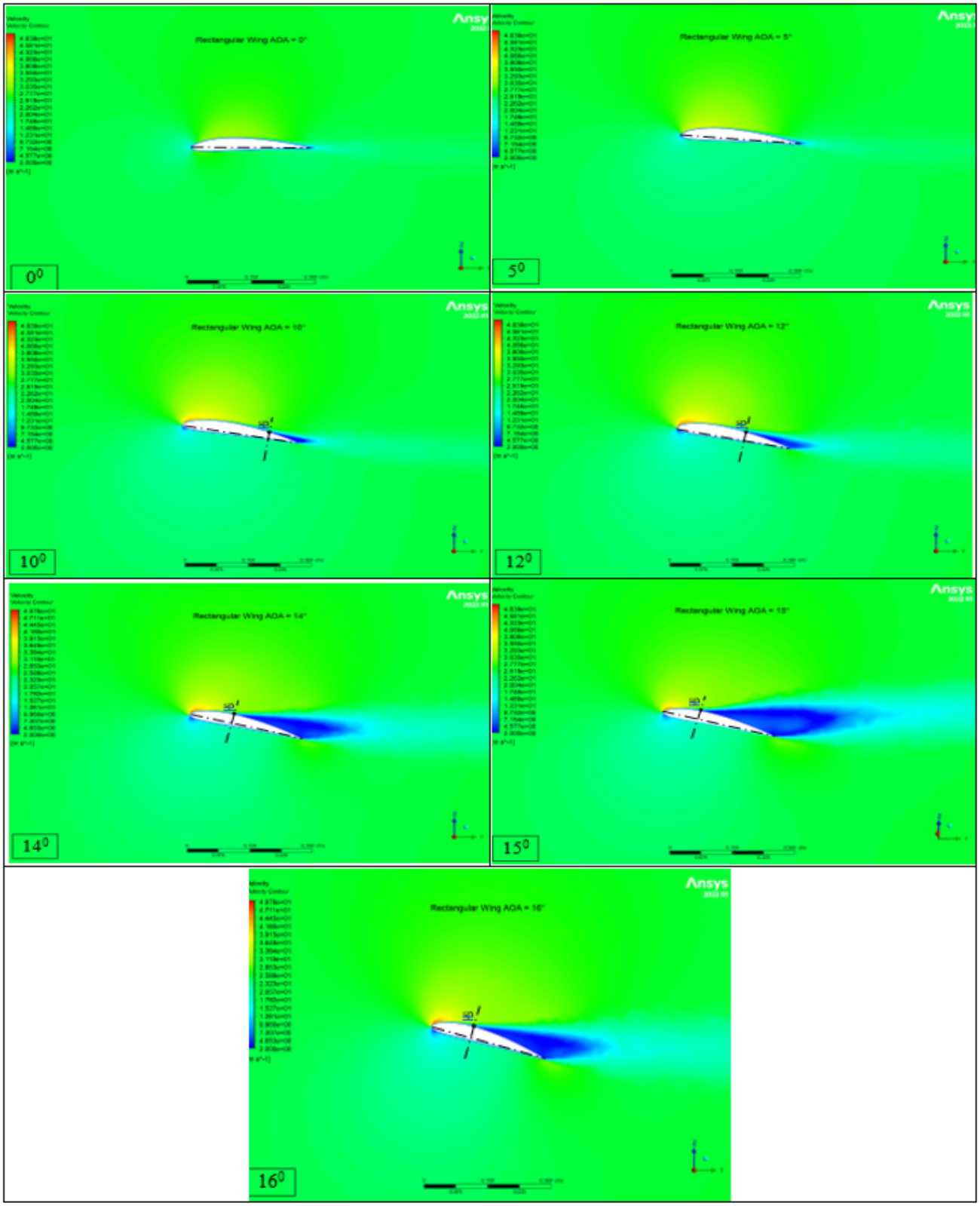


Figure 3.4: Velocity contours of rectangular wing at $0-5-10-12-14-15-16^{\circ}$ of angles of attack

3.4. Vortices around wings

Prominent vortex patterns around the wing and along the direction of airflow were illustrated in Figures 3.7, 3.8 and 3.9. Interactions between the flow and the wing resulted in the wing being enveloped by numerous small vortices. Notably, there's an absence of vortex development at the wing's trailing edge, where the airflow is disrupted by the wing structure. Vortex generation at the wingtips occurred at every angle of attack. For a rectangular wing, the wingtip vortex significantly affects the surrounding area once the angle of attack exceeds 5°. The trapezoidal wing exhibits the least amount of wingtip vortex. As the angle of attack increases, the vortices on the wing's upper surface become more chaotic, leading to their detachment from the wing's surface sooner, before they can reach the wing's rear. At attack angles

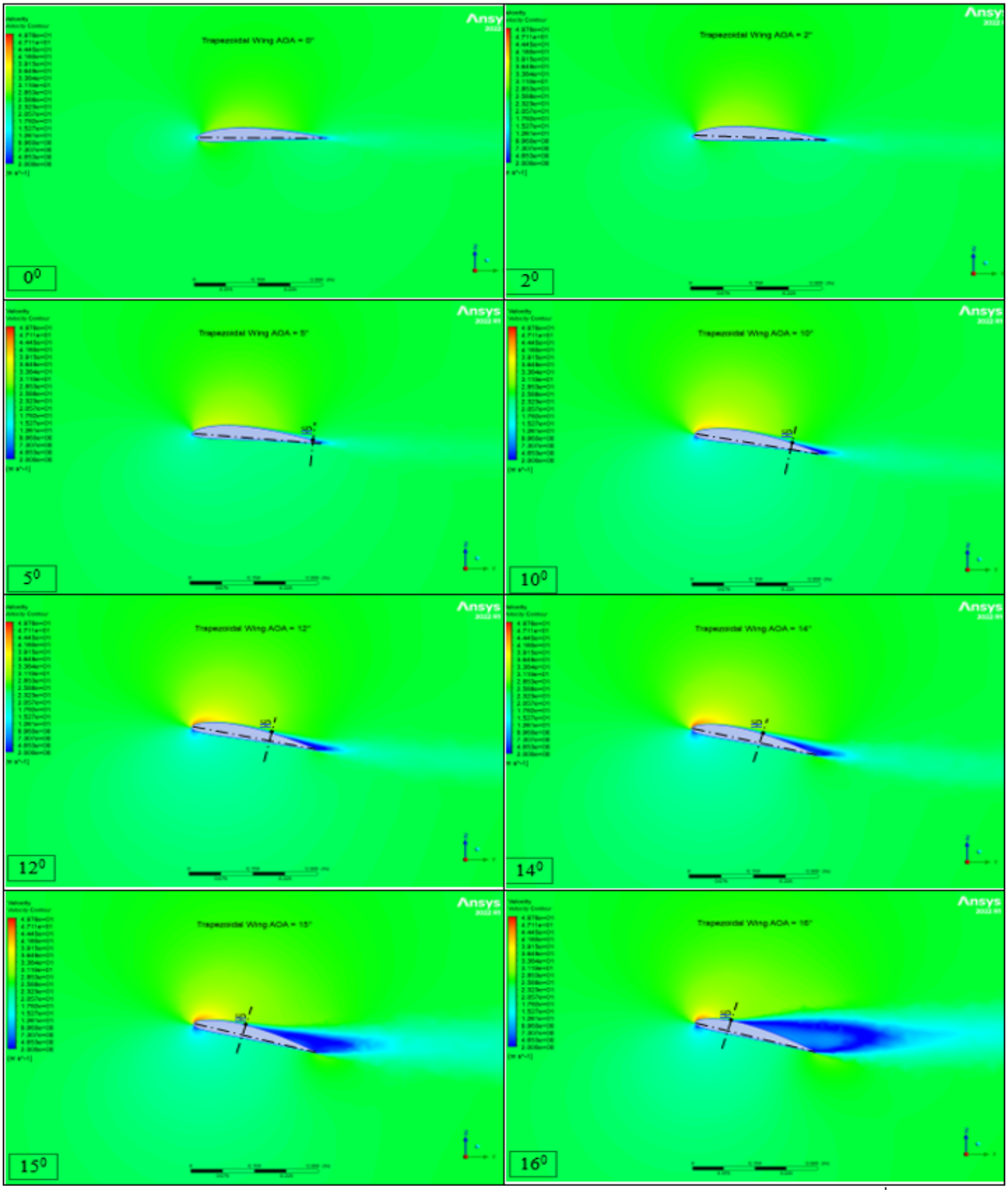


Figure 3.5: Velocity contours of trapezoidal wing at $0-2-5-10-12-14-15-16^{\circ}$ of angles of attack

of 0, 2, and 5°, the wing surface vortices were formed in a stable manner. When the angle of attack reached 10°, there were noticeable disturbances in the flow along the entire length of the wing, particularly at the trailing edge. At an angle of attack of 12°, the rectangular wing's surface showed minor disruptions due to vortex merging, a phenomenon not yet seen on trapezoidal and arrow-angle wings. At 14°, the minor disturbances on the rectangular wing's surface became more pronounced. The trapezoidal wing began to exhibit small to medium-sized vortex disruptions, and the arrow-angle wing started to show small-scale vortex disturbances. At 15° angle of attack, the rectangular wing experienced the formation of a significant vortex, while the trapezoidal wing had two vortices of medium size, and the arrow-angle wing developed small to medium vortices spread across its surface. For the rectangular and trapezoidal wings at a 16° angle of attack, large vortices were present on the wing surfaces, which substantially disrupted and destabilized the airflow. However, on the arrow-angle wing at attack angles of 17° and 18°, the flow irregularity was minimized near the junction of the wing and the *UAV* body. Conversely, on the other wing types, the connection point to the fuselage showed increased irregularity and larger vortex formations.

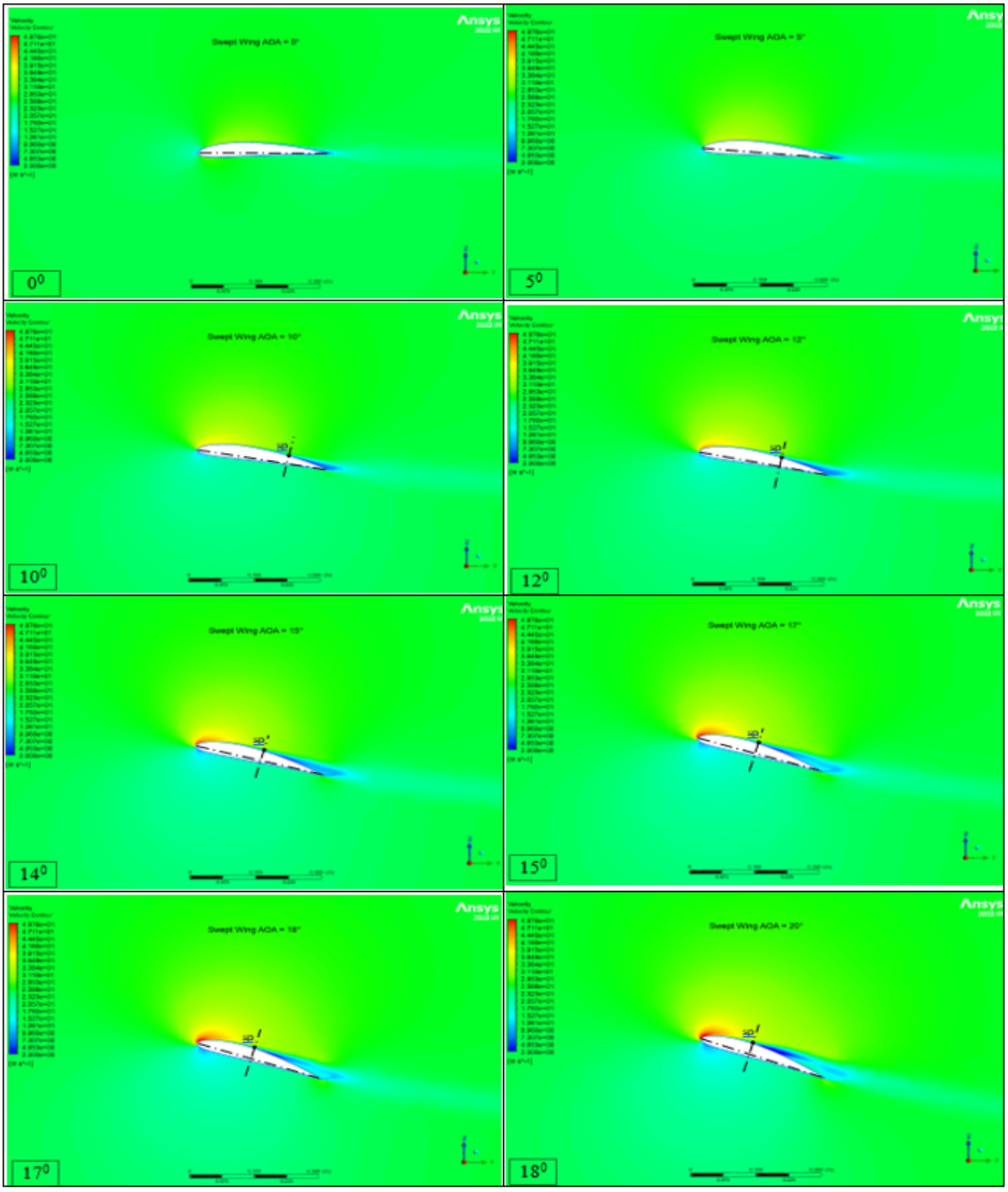


Figure 3.6: Velocity contours of arrow angle wing at $0-5-10-12-14-15-17-18^{\circ}$ of angles of attack

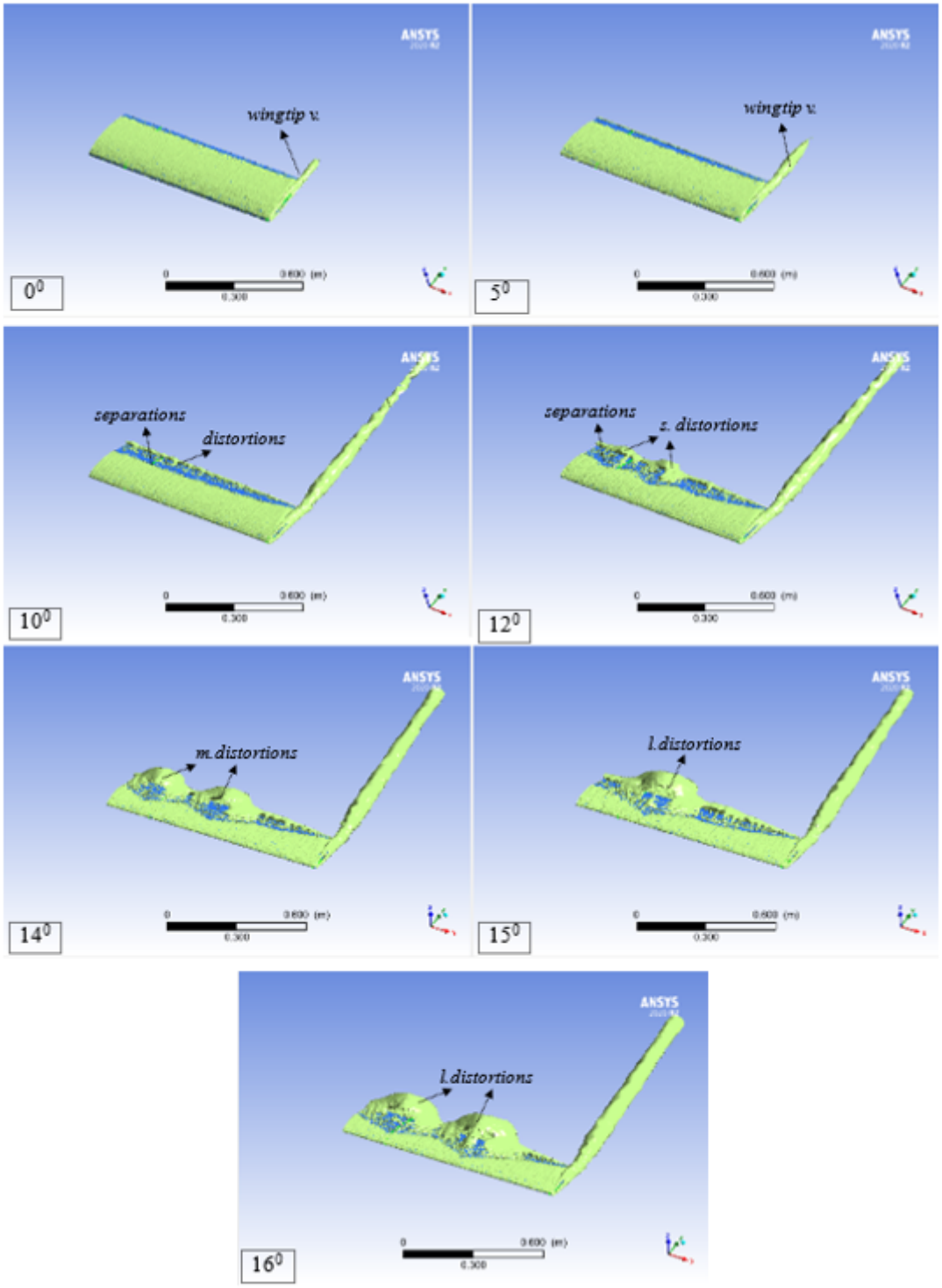


Figure 3.7: Vortex formations of rectangular wing at $0 - 5 - 10 - 12 - 14 - 15 - 16^{\circ}$ of angles of attack

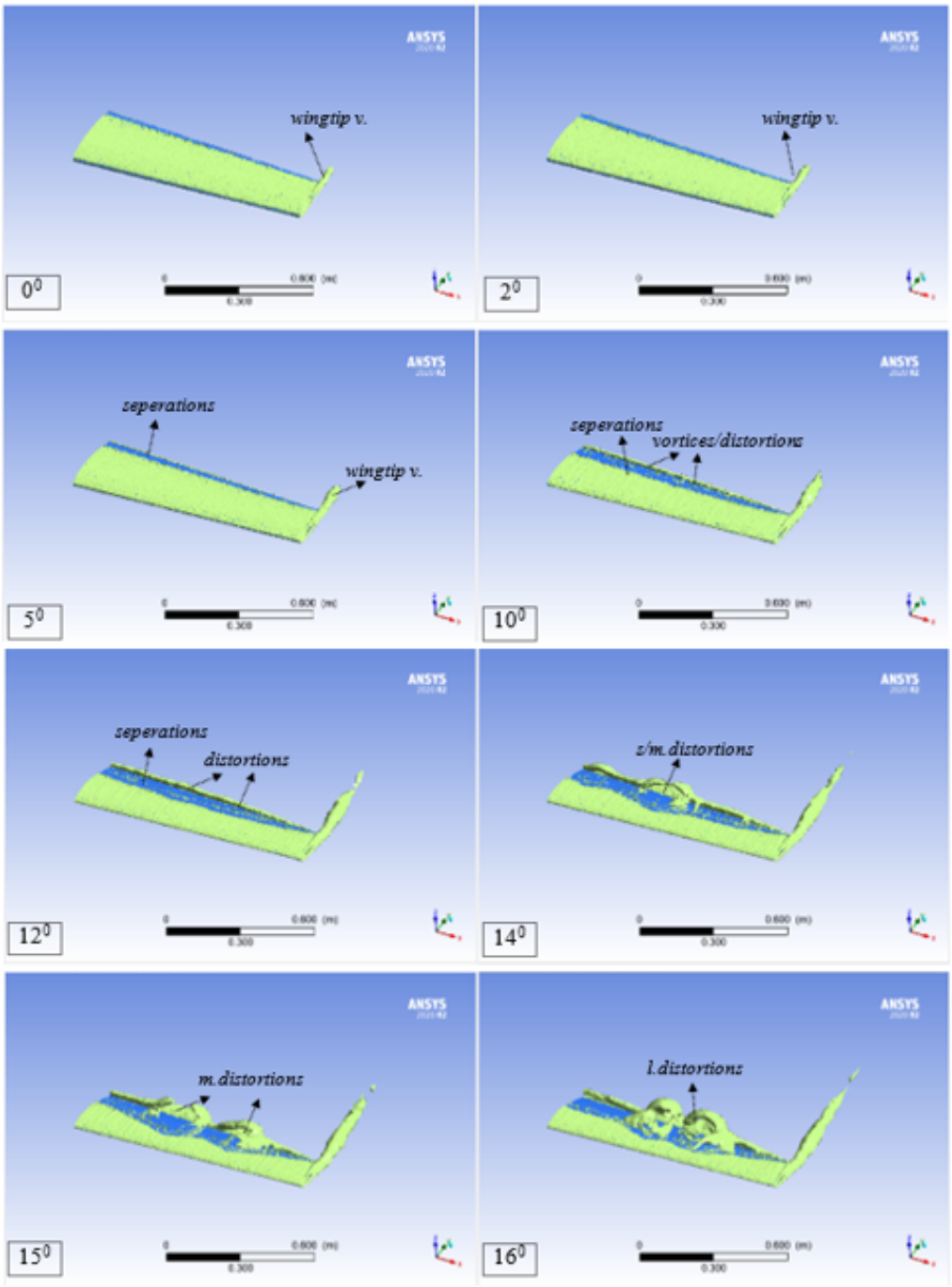


Figure 3.8: Vortex formations of trapezoidal wing at $0-2-5-10-12-14-15-16^{\circ}$ of angles of attack

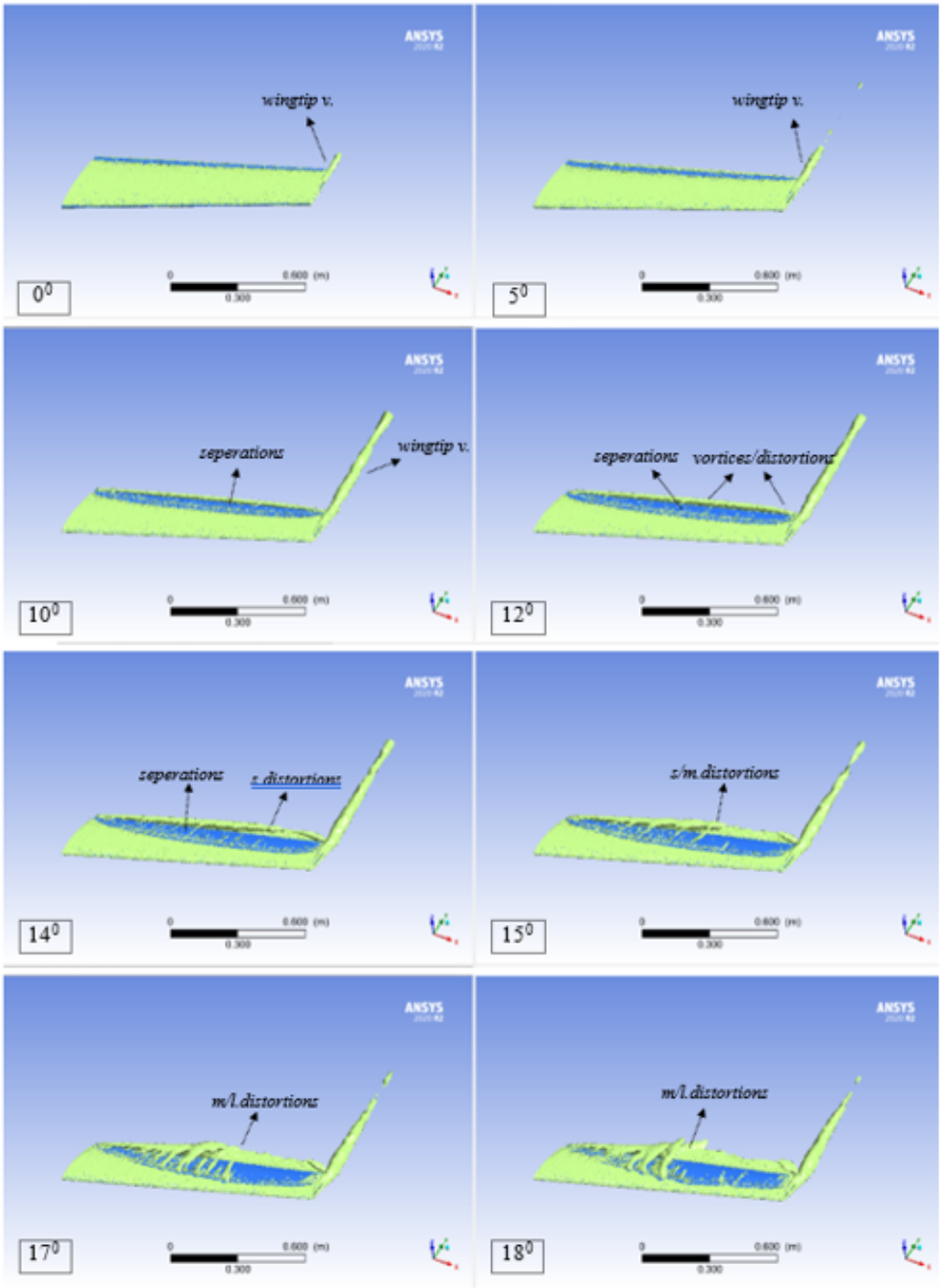


Figure 3.9: Vortex formations of arrow angle wing at $0-5-10-12-14-15-17-18^{\circ}$ of angles of attack

3.5. Pressure behaviour of the body and tail wing

Flow analysis conducted on the body revealed that it generates a slight lift. The lift coefficient (C_l) and drag coefficient (C_d) were found to be 0.00015200737 and 0.0084482443, respectively. Figure 3.10 illustrates the pressure contours for the body alone. The body's basic cylindrical shape and the tapered design at both ends ensure even pressure distribution on and around the body, with minimal pressure variation. Consequently, the wings are responsible for generating the necessary lift force. The maximum pressure was observed at the front part of the body, specifically at the nose of the fuselage.

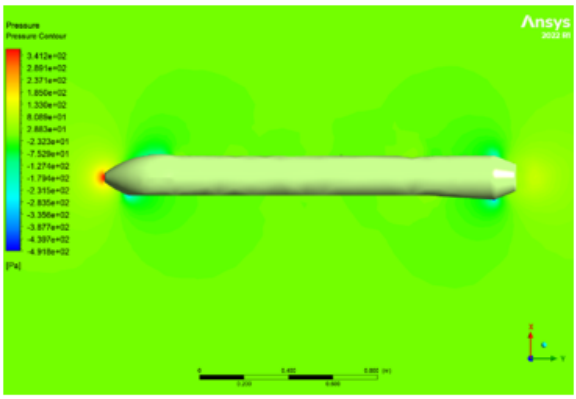


Figure 3.10: Pressure contours from the middle of fuselage

Figure 3.11 depicts the tail wings, which are designed with a distinct drop section model compared to the main wings. The profile's geometry results in high pressure exclusively at the front wing's leading edges where the airflow first impacts. The rest of the wing and its trailing edges experience a consistent pressure distribution. The V-shaped tail is confirmed to produce a measurable lift effect.

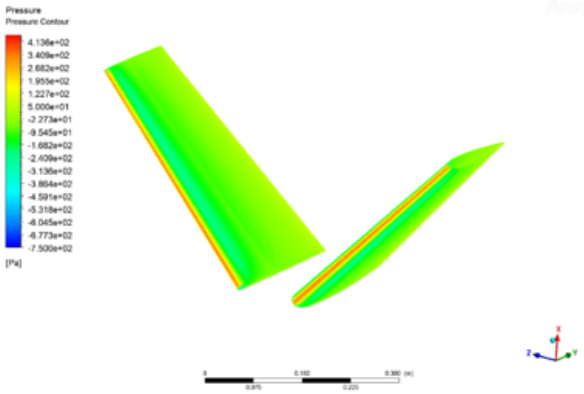


Figure 3.11: Tail pressure distribution

3.6. Effect of wing layout on lift and drag

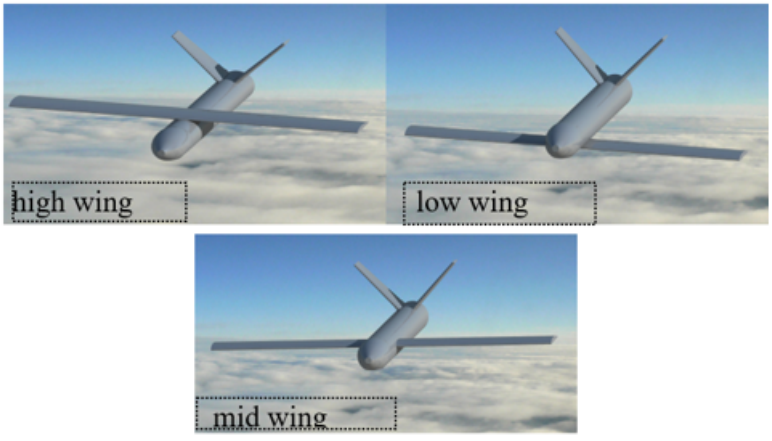
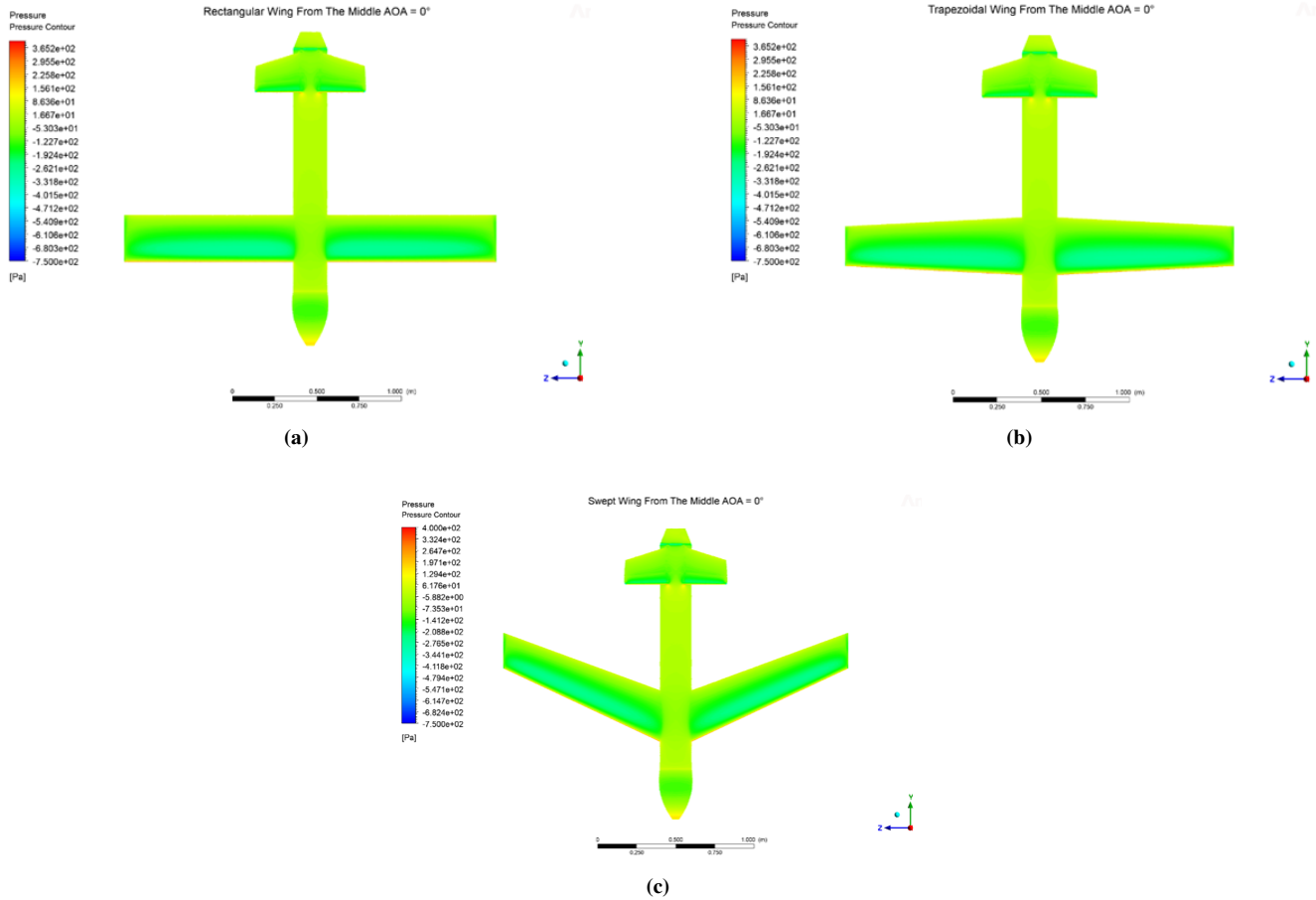


Figure 3.12: Location of the wing on the fuselage

The point where the wings attach to the main body, or fuselage, is crucial. Figure 3.12 displays three different wing attachment locations: high, mid, and low. Table 3.4 lists the lift (C_l) and drag (C_d) coefficients resulting from these wing-fuselage configurations. The arrow-angle from below model showed the highest lift coefficient, while the trapezoidal from the middle model had the best C_l/C_d ratio. Thus, the trapezoidal middle model is the preferred choice. Based on this, Figure 3.13 presents the pressure outcomes for rectangular, trapezoidal, and arrow-angle wings when attached mid-fuselage. Since the fuselage and tail remain unchanged, the wings show no significant pressure differences. The areas of highest pressure were identified at the nose of the fuselage, the leading edges of the wings, and where the tail bonds the fuselage. Conversely, the lowest pressure was noted at the front of the wing's upper surface camber. This area of low pressure is also visible in Figures 3.1, 3.2, and 3.3, which depict earlier pressure results.

Table 3.4: C_d and C_l values according to the connection position of the wing

Juncture	C_l	C_d	C_l/C_d
Rectangle from top	0.34433	0.038805	8.87334
Rectangle from the middle	0.37425	0.0381	9.82292
Trapezoidal from the middle	0.378266	0.03782	10.00121
Trapezoidal from below	0.2784	0.03939	7.06778
Arrow-angle from middle	0.35411	0.03777	9.37481
Arrow-angle from below	0.39897	0.0384	7.78567



(a) **(b)** **(c)**

Figure 3.13: Mid-Wing Rectangular, Trapezoidal, Arrow-Angle Pressure Distribution

3.7. Flight-cost calculation of Kamikaze *UAV* models

Based on ideal conditions, a basic flight-cost calculation for Kamikaze *UAVs* has been examined. Some assumptions and used values in the examinations made for rectangular and trapezoidal wings;

- Optimization values $C_{D\,\text{rectangular wing}} = 0.02089$
- Optimization values $C_{D\,\text{arrow-angle wing}} = 0.01976$
- The price of the fuel used, Kerosene, is 1 liter for Turkey = 1.154 dollar
- Kerosene fuel density $\rho = 0.82\,kg/L$
- Kerosene calorific value $(ID) = 46\,MJ/kg$
- Aircraft wing top view area $S = 0.55\,m^2$
- $V_s = 25\,m/s$, $\rho_{air} = 1.225\,kg/m^3$

• Aircraft engine efficiency $\eta = 85\%$
• Fuel calculation in dollars for 1 year of rectangular wing flight time;

$$\begin{aligned} F_D &= (1/2)\,\rho.V_\infty^2.A.C_{D\,\text{rectangular wing}} \\ F_D &= (1/2)\times 1.225\times 25^2\times 0.02089 = 4.3997 \\ W_D &= F_D.V_\infty = 4.3997\left(\frac{kg.m}{s^2}\right)\times 25\left(\frac{m}{s}\right) = 109.9925\frac{kg.m^2}{s^3} \qquad (3.1) \\ W_{motor} &= \frac{W_d}{\eta} = \frac{109.9925}{0.85} = 129.403\left(\frac{kg.m^2}{s^3}\right) \qquad (3.2) \\ W_{motor} &= \dot{m}.ID = \rho\,.\dot{\forall}\,.ID \\ \dot{\forall} &= \frac{W_{motor}}{\rho_{fuel}.ID} = \frac{129.403\,\frac{kg.m^2}{s^3}}{0.82\,\frac{kg}{L}.46\frac{MJ}{kg}} = 3.431\times 10^{-6}\frac{L}{s} \\ \dot{\forall} &= 3.431\times 10^{-6}\frac{L}{s}.3600\frac{s}{h} = 0.0124\frac{L}{h} \end{aligned}$$

for 8765 hours;

$$\begin{aligned} 0.0124\frac{L}{h}\times 8765\,h &= 108.686\,L \\ 108.686\times 1.154 &= 125.424\text{ dollar} \end{aligned}$$

Fuel calculation in dollars for 1 year of rectangular wing flight time;

$$\begin{aligned} F_D &= (1/2)\,\rho.V_\infty^2.A.C_{D\,\text{arrow}-\text{angle wing}} \\ F_D &= (1/2)\times 1.225\times 25^2\times 0.01976 = 4.1604 \\ W_D &= F_D.V_\infty = 4.1604\left(\frac{kg.m}{s^2}\right)\times 25\left(\frac{m}{s}\right) = 104.01\frac{kg.m^2}{s^3} \\ W_{motor} &= \frac{W_d}{\eta} = \frac{104.01}{0.85} = 122.365\left(\frac{kg.m^2}{s^3}\right) \\ W_{motor} &= \dot{m}.ID = \rho\,.\dot{\forall}\,.ID \\ \dot{\forall} &= \frac{W_{motor}}{\rho_{fuel}.ID} = \frac{122.365\,\frac{kg.m^2}{s^3}}{0.82\,\frac{kg}{L}.46\frac{MJ}{kg}} = 3.244\times 10^{-6}\frac{L}{s} \\ \dot{\forall} &= 3.244\times 10^{-6}\frac{L}{s}.3600\frac{s}{h} = 0.0117\frac{L}{h} \end{aligned}$$

for 8765 hours;

$$\begin{aligned} 0.0117\frac{L}{h}\times 8765\,h &= 102.5505\,L \\ 102.5505\times 1.154 &= 118.343\text{ dollar} \end{aligned}$$

Arrow-angle kamikaze *UAV* was determined as the most suitable in flight-cost comparison.

4. Conclusions and Recommendations

In this study, the effects of wingspan shapes and wing-fuselage junction on the aerodynamic performance of an unmanned kamikaze aircraft were investigated. Three different wingspan models were used: trapezoidal, rectangular and arrow-angle shapes. The effects of angle of attack on the aerodynamic performance of different modelled unmanned aerial vehicles are shown. Pressure/velocity profiles and vortex formation results were obtained and the differences in the results were compared. C_l and C_d coefficients were obtained with the computational fluid mechanics analysis. The main findings found in the study can be summarized as follows.
Considering the pressure and velocity profiles and vortex formations, three different wing configurations were examined for kamikaze *UAVs* and the results of the effect of the angle of attack were given. C_l and C_d coefficients were obtained with the computational fluid mechanics analysis. The main findings found in the study can be summarized as follows;

- The structure of the pressures around the wings started to differ as a result of 15°. At 15°, the L– and L- negative pressure zones above the wing started to decrease in the rectangular and trapezoidal wing, while they increased in the arrow-angle wing.
- In the arrow-angle wing, according to other results, SP occurred at the rear and a low velocity profile without intense was formed.
- The lowest wingtip vortex formation occurred in the trapezoidal wing. When the vortex formations are examined, the arrow-angle wing structure creates a more appropriate distribution.

- When the location of the wing mounting to the fuselage and the type of wing were investigated, trapezoidal from the mid showed the most appropriate C_l/C_d performance.
- The trapezoidal wing has the highest lift coefficient up to 15° *AoA*. After 15°, the lift coefficient of both models started to decrease with the rectangular wing.
- While the arrow-angle wing exhibited the lowest lift coefficient behavior, the lift coefficient increased continuously up to 20°. There was no decrease in the lift coefficient.
- The drag coefficient increased in all models. The arrow-angle wing model has the lowest drag coefficient.
- The C_l/C_d ratio increased in all models up to 5° *AoA*. After 5° *AoA*, this ratio decreased in all wing models. While the trapezoidal wing has the highest ratio up to 14 ° AoA, the C_l/C_d ratio of the arrow angle model is higher at higher angles of attack.

While it is logical to use the trapezoidal wing up to 15° *AoA*. it is more appropriate to choose the arrow-angle wing model, which still increases the lift coefficient at higher angles of attack.

Article Information

Acknowledgements: The authors would like to express their sincere thanks to the editor and the anonymous reviewers for their helpful comments and suggestions..

Author's contributions: All authors contributed equally to the writing of this paper. All authors read and approved the final manuscript.

Conflict of interest disclosure: No potential conflict of interest was declared by the authors.

Supporting/Supporting organizations: No grants were received from any public, private or non-profit organizations for this research.

Ethical approval and participant consent: It is declared that during the preparation process of this study, scientific and ethical principles were followed and all the studies benefited from are stated in the bibliography.

Plagiarism statement: This article was scanned by the plagiarism program.

References

[1] J. Karimi, S. H. Pourtakdoust, *Optimal maneuver-based motion planning over terrain and threats using a dynamic hybrid PSO algorithm*, Aerosp. Sci. Technol., **26**(1) (2013), 60-71.
[2] J. P. Škrinjar, P. Škorput, M. Furdić, *Application of unmanned serial vehicles in logistic processes*, In New Technologies, Development and Application 4, Springer, 2019, 359-366.
[3] V. Hassija, et al., *A survey on IoT security: Application areas, security threats, and solution architectures*, IEEE Access, **7** (2019), 82721-82743.
[4] Y. Unpaprom, N. Dussadeeb, R. Ramaraj, *Modern Agriculture Drones, Modern Agriculture Drones Chapter: Modern Agriculture Drones the Development of Smart Farmers*, 2018, 13-19.
[5] I. Jeelani, M. Gheisari, *Safety challenges of UAV integration in construction: Conceptual analysis and future research roadmap*, Safety Science, **144** (2021), 105473.
[6] O. Adepoju, et al., *Drone/unmanned aerial vehicles (UAVs) technology, Re-skilling Human Resources for Construction 4.0: Implications for Industry*, Academia and Government, (2022), 65-89.
[7] F. Zeng, *Nested vehicle routing problem: Optimizing drone-truck surveillance operations*, Trans. Res. Part C: Emerging Tech., **139** (2022), 103645.
[8] P. Garg, et al., *Isdnet: Ai-enabled instance segmentation of aerial scenes for smart cities*, ACM Trans. Internet Tech. (TOIT), **21**(3) (2021), 1-18.
[9] A. Restas, *Drone applications for supporting disaster management*, World Journal of Engineering and Technology, **3**(3) (2015), 316-321.
[10] A. Straubinger, H.L. de Groot, E. T. Verhoef, *E-commerce, delivery drones and their impact on cities*, Transportation Research Part A: Policy and Practice, **178** (2023), 103841.
[11] H. Shakhatreh, et al., *Unmanned aerial vehicles (UAVs): A survey on civil applications and key research challenges*, IEEE Access, **7** (2019), 48572-48634.
[12] S. A. Hoseini, et al. *Trajectory optimization of flying energy sources using q-learning to recharge hotspot UAVs*, in IEEE INFOCOm 2020-IEEE conference on computer communications workshops (INFOCOm WKSHPS), 2020, IEEE, 683-688.
[13] M. Prieto, M.S. Escarti-Guillem, S. Hoyas, *Aerodynamic optimization of a VTOL drone using winglets, Results in Engineering*, **17** (2023) 100855.
[14] V. Chamola, et al., *A comprehensive review of the COVID-19 pandemic and the role of IoT, drones, AI, blockchain, and 5G in managing its impact*, IEEE Access, 8 (2020), 90225-90265.
[15] K. Li, et al., *Energy efficient legitimate wireless surveillance of UAV communications*, IEEE Transactions on Vehicular Technology, **68**(3) (2019), 2283-2293.
[16] G. Bai, et al., *Network approach for resilience evaluation of a UAV swarm by considering communication limits*, Reliability Engineering & System Safety, **193** (2020), 106602.
[17] S. R. Edulakanti, S. Ganguly, *The emerging drone technology and the advancement of the Indian drone business industry*, The Journal of High Technology Management Research, **34**(2) (2023), 100464.
[18] S.G. Kontogiannis, J.A. Ekaterinaris, *Design, performance evaluation and optimization of a UAV*, Aerosp. Sci. Technol., **29**(1) (2013), 339-350.
[19] C. Fu, et al., *Adaptive robust backstepping attitude control for a multi-rotor unmanned aerial vehicle with time-varying output constraints*, Aerosp. Sci. Technol., 78 (2018), 593-603.
[20] P. Panagiotou, K. Yakinthos, *Aerodynamic efficiency and performance enhancement of fixed-wing UAVs*, Aerosp. Sci. Technol., 99 (2020), 105575.
[21] J. Anderson, *EBOOK: Fundamentals of Aerodynamics (SI units)*, McGraw Hill, 2011.
[22] A. Quintana, et al., *Aerodynamic analysis and structural integrity for optimal performance of sweeping and spanning morphing unmanned air vehicles*, Aerosp. Sci. Technol., **110** (2021), 106458.
[23] M. Voskuijl, *Performance analysis and design of loitering munitions: A comprehensive technical survey of recent developments*, Defence Technology, **18**(3) (2022), 325-343.
[24] D. Zampronha, A. Albuquerque, *Cheaper Precision Weapons: An Exploratory Study about the HESA Shahed 136*, Advances in Aerosp. Sci. Technol., **9**(1) (2024), 40-59.
[25] T. Saraçyakupoğlu, H. D. Delibaş, A. D. Özçelik, *An experimental determination and numerical analysis of a loiter munition unmanned aerial vehicle system*, International Journal of 3D Printing Technologies and Digital Industry, **6**(1) (2022), 83-101.
[26] E. Sakarya, A. Alkan, *Savunma sanayiinde kullanilabilecek Kamikaze İha uygulaması*, Bilgisayar Bilimleri ve Teknolojileri Dergisi, **2**(1) (2021), 24-28.
[27] A. Sadikin, et al., *A comparative study of turbulence models on aerodynamics characteristics of a NACA0012 airfoil*, International Journal of Integrated Engineering, **10**(1) (2018).
[28] C. Suvanjumrat, *Comparison of turbulence models for flow past NACA0015 airfoil using OpenFOAM*, Engineering J., **21**(3) (2017), 207-221.