J. Phys. Res. Edu., Vol. 2, March 2025

Estimating Mass and Luminosity of Spectroscopic Binary Stars Using a Python-Based Computational Approach

Jayashree Kundu[1], Rakesh Kumar Mandal[1], and Tamal Sarkar[2]*
[1]*Dept. of Computer Science & Technology, University of North Bengal, Siliguri, 734013, India*
[2]*HECRRC, University of North Bengal, Siliguri, 734013, India*

Spectroscopic binary stars provide crucial insights into stellar masses, orbital dynamics, and evolutionary processes. This study presents a computational approach to analyzing spectroscopic binary systems using Python by developing an algorithm to estimate their mass and luminosity. In our study, the algorithm processes the spectral data, particularly variations in the Hydrogen-alpha line over time (in Modified Julian Date), to compute radial velocities by incorporating Doppler shifts and applying barycentric corrections. A sinusoidal function is then fitted to the velocity variations to determine the orbital period. The stellar masses are derived using the radial velocity curve with the inclination angle and orbital parameters. Given the mass-luminosity relation, the luminosities of the stars are estimated. Since the derived mass of the system ranges between 2 and 55 times the mass of the sun, the luminosity can be calculated based on the mass-luminosity relation, where luminosity is proportional to the stellar mass raised to the power of 3.5. This computational method offers an efficient and accurate technique for studying spectroscopic binaries, which can be extended to analyze large datasets from astronomical surveys, enhancing our understanding of binary star evolution.

I. INTRODUCTION

In the context of astronomy, the term *binary* was first used by Sir William Herschel in 1802. Later, american astronomer and physicist Edward Charles Pickering and Carl Vogel discovered the first spectroscopic binary stars in November 1889. These stars are bound by gravity and orbit each other around a common center of mass, detectable only through spectroscopy. Binary stars, particularly spectroscopic binaries, have long been a cornerstone in stellar astrophysics because of their importance in determining fundamental stellar parameters such as mass and luminosity. Historical and modern studies have utilized radial velocity variations caused by Doppler shifts to infer the orbital dynamics of these systems. By linking observational data with theoretical models, astronomers have constructed essential relations, such as the mass-luminosity relation, which is pivotal for understanding stellar structure and evolution. With the advent of modern computational tools and the increasing availability of high-quality spectroscopic data, the development of automated and precise algorithms has become essential. This work aims to contribute to this field by developing a Python-based algorithm to extract mass and luminosity estimates from spectroscopic binary observations, enhancing the reliability and efficiency of such analyses.

The study of spectroscopic binaries has significantly evolved over the decades, with numerous researchers contributing to the field through innovative methodologies and refined observational techniques. In 1941, Wilson pioneered a method for determining mass ratios in spectroscopic binaries requiring substantially fewer observational data points than a full orbital solution. This research represented a significant advancement in the mass determination of double-lined spectroscopic binaries (SB2s), even when a complete orbital computation was not feasible. Wilson's approach provided astronomers with a more efficient technique for estimating mass ratios, thus facilitating the characterization of a greater number of binary systems [1]. Subsequent research by Morbey and Brosterhus in 1974 expanded on earlier methodologies by focusing on the identification of spectroscopic binaries from previously published radial velocity datasets. Their work emphasized the importance of statistical analysis in recognizing periodic velocity variations, which are indicative of binary motion. By leveraging pre-existing observational data, their study demonstrated the potential of archival research in discovering new binary systems without the need for extensive new observational campaigns [2]. A significant study in spectroscopic binary mass determination came with the work of Zucker and Alexander in 2006, who introduced relativistic effects into the analysis of spectroscopic binaries. Their study incorporated special relativistic corrections into the mass function calculations, thus refining mass estimates, particularly for binaries exhibiting high radial velocities. The incorporation of relativistic effects provided a more accurate framework for understanding the orbital dynamics of massive and compact binary systems, which had previously been modelled using Newtonian mechanics [3]. In 2020, Heyne contributed to the field by conducting a spectroscopic characterization of nine binary star systems, including HIP 107136 and HIP 107533. His study utilized high-resolution spectroscopy to determine stellar parameters such as effective temperature, surface gravity, and metallicity. The research also provided updated radial velocity curves, contributing valuable data to the ongoing efforts in binary star classification and orbital modelling [4]. More recently, Kounkel et al., in 2021, conducted an extensive survey of double-lined spectroscopic binaries using data from the APOGEE DR16 and DR17 datasets. Their large-scale analysis leveraged machine learning techniques to identify and classify binary stars within the dataset. The study highlighted the importance of integrating big data methodologies with traditional spectroscopic analyses, allowing for the rapid and precise characterization of binary star populations on a scale previously unattainable [5].

These works collectively illustrate the progression of spectroscopic binary research from early manual calculations to modern automated and computationally intensive techniques. Despite significant advancements in the study of spectroscopic binary stars over the past century, there are notable gaps and challenges in the automation and precision of mass and luminosity determination. Traditional methods heavily rely on manual data analysis and simplified models, which may not fully exploit the potential of modern high-precision spectroscopic datasets. Additionally, many existing algorithms either lack barycentric corrections, assume circular orbits, or do not provide flexibility for handling large-scale survey data. Furthermore, while some computational methods exist, they are often limited in accessibility, scalability, or ease of use for astronomers working with diverse observational datasets. The integration of robust, automated algorithms-capable of handling barycentric corrections, Doppler shift analysis, orbital fitting, and mass-luminosity estimations-remains an underdeveloped area. Therefore, this research addresses the need for a user-friendly, Python-based algorithm that can accurately process spectroscopic binary data, automate essential calculations, and adapt to the increasing volume and complexity of astronomical data from modern surveys, as it has become one of the most popular programming languages in astronomy due to its versatility, ease of use, and rich ecosystem of scientific libraries for astronomical research [6, 7]. Some prominent ones include NumPy for numerical computations, SciPy for scientific computing, matplotlib for graphical representation, and Astropy for astronomical data analysis. Another important parameter for Python is that it is an open-source language that aligns well with the principles of openness and transparency in scientific research. This allows researchers to inspect, modify, and share the code freely [8]. Python is an essential tool in computational astrophysics, allowing astrophysicists to analyze data efficiently, model physical systems, and visualize results [9]. Python frameworks like Scikit-learn, TensorFlow, and PyTorch enable the prediction of stellar properties and analysis of complex datasets [10]. In the following section, we discuss computational code development with the help of Python libraries [11]. This paper has seven sections: Section II-Objectives, Section III-Data Collection, Section IV-Methodology, Section V-Results Analysis, Section VI-Discussion, Section VII-Conclusion, and Annexure-A, which describes the formulation used for developing the codes.

II. OBJECTIVES

The primary objective of this study is to develop an efficient and accurate computational method for analyzing spectroscopic binary systems, with a focus on determining stellar masses and luminosities. Specifically, this research aims to:

(i) Develop a Python-based algorithm for processing spectroscopic binary data,

(ii) Calculate the individual masses of binary star components using radial velocity curves and orbital parameters,

(iii) Estimate stellar luminosities using the established mass-luminosity relation,

(iv) Ensure the scalability and adaptability of the algorithm for analyzing large datasets from astronomical surveys.

III. DATA COLLECTION

We are using secondary data from Smriti Mahajan's book. Here, the wavelengths for Hydrogen-alpha lines as observed for the two components of a star system at different times have been retrieved [12].

Sl. No.	Time (Julian date)	Line1 (in Å)	Line2 (in Å)
1	243866.01	6564.114	6562.15
2	243866.09	6564.48	6561.41
3	243866.18	6564.777	6560.811
4	243866.26	6564.984	6560.392
…	…	…	…
⋮	⋮	⋮	⋮
22	243867.76	6562.865	6564.676
23	243867.84	6563.279	6563.839
24	243867.93	6563.706	6562.977
25	243868.01	6564.116	6562.148

TABLE I: Data for the spectroscopic binary system of Hydrogen Alpha lines.

IV. METHODOLOGY

The methodology for this study is structured into seven systematic steps:

A. Preparing Dataset

We construct a dataset Table I with four columns that include the serial number, observation time in Julian Date, and Hydrogen-alpha spectral line measurements (in Angstroms) for each component of the binary system.

B. Reading input files in Python Environment

The observational data from Table I is imported into a Python environment using the following command:

```
import pandas as pd
df = pd.read_excel('Binary_Star_System_Data.xlsx')
```

C. Calculating velocities of curve

We compute the radial velocities of each spectral line using the Doppler effect formula [13], with a barycentric correction to account for Earth's motion away from the observed object:

```
L = len(arr_Time)
for i in range(0,L):
    Lambda1 = arr_Line1[i]
    Lambda2 = arr_Line2[i]
    v1_BC = c*(( Lambda1- Lambda0)/Lambda0) - BC
    v2_BC = c*(( Lambda2- Lambda0)/Lambda0) - BC
    v1_BC_list.append(v1_BC)
    v2_BC_list.append(v2_BC)
df['Velocity1_BC'] = v1_BC_list
df['Velocity2_BC'] = v2_BC_list
```

```
# Finding the Average velocity for stars.
Star1_v = (np.abs(np.max(v1_BC_list)) + np.abs(np.min(v1_BC_list)))/2
Star2_v = (np.abs(np.max(v2_BC_list)) + np.abs(np.min(v2_BC_list)))/2
```

The average velocities for each star are then calculated from the peak and trough values of the velocity lists.

D. Determining the Time-Period of the Binary system

The curve/curves show a sinusoidal nature. We define a sinusoidal model function to fit the velocity data:

```
def Spectral_velo_f(x,A,f,P,V):
    return (A*np.sin(f*(x-P))+V)
```

Optimal values for amplitude, frequency, phase shift, and vertical offset are estimated using curve_fit from the *scipy.optimize* library. After generating interpolated velocity curves, we compute the first and second derivatives to find critical points (maxima and minima). The average interval between critical points is doubled to determine the full orbital period. Now, the period of the Binary system is determined by the following steps.

(i) Finding the Optimal values of curves

We find the optimal value of Amplitude, frequency, Phase shift, and Vertical shift by guessing the initial values to fit the curve through the following coding.

```
# Importing required packages.
from scipy.optimize import curve_fit

# For Curve-1.
popt1,pcov1 = curve_fit(Spectral_velo_f,x_data,y1_data,
    p0=[-75,0.251,243866.01,0])
A1_opt,f1_opt,P1_opt,V1_opt = popt1
```

```
    # For Curve-2.
    popt2,pcov2 = curve_fit(Spectral_velo_f,x_data,y2_data,
        p0=[150,0.251,243866.01,0])
    A2_opt,f2_opt,P2_opt,V2_opt = popt2
```

(ii) Fitting the curves

```
# Importing required packages
from sympy import symbols, sin

# Creating extra points to fit the curves.
x_new = np.linspace(243866.01,243868.01,50)
for i in range(len(x_new)):
    # 1st curve.
    y1 = Spectral_velo_f(x_new[i],A1_opt,f1_opt,P1_opt,V1_opt)
    # 2nd curve.
    y2 = Spectral_velo_f(x_new[i],A2_opt,f2_opt,P2_opt,V2_opt)
    y1_new.append(y1)
    y2_new.append(y2)

# Fitted curves.
x = symbols(’x’)
Spectral_f1 = -round(A1_opt,2)*sin(f1_opt*(x-P1_opt))+V1_opt
Spectral_f2 = round(A2_opt,2)*sin(f2_opt*(x-P2_opt))+V2_opt
```

(iii) Finding the Maxima-Minima of the curves

By running the following piece of code, we can find the Maxima and Minima of both the curves through applying the Differential Calculus. After getting the values for $\frac{dy}{dt}$ and $\frac{d^2y}{dt^2}$, we have found its critical points.

```
# Importing required packages
from sympy import symbols, diff, solve
```

```
#Compute the first-order and the second-order derivatives.
f1_prime = diff(Spectral_f1, x)
f2_prime = diff(Spectral_f2, x)
f1_double_prime = diff(f1_prime, x)
f2_double_prime = diff(f2_prime, x)

#Substituting the value of First-order and Second-order derivatives.
for i in range(len(x_new)):
    f1 = f1_prime.subs(x,x_new[i])
    f2 = f2_prime.subs(x,x_new[i])
    f1_2= f1_double_prime.subs(x,x_new[i])
    f2_2 = f2_double_prime.subs(x,x_new[i])
    L1.append(f1)
    L2.append(f2)
    L1_2.append(f1_2)
    L2_2.append(f2_2)

#Computing the critical points.
for i in range(len(df_temp)):
    f1_prime_temp = df_temp['f1_prime'].iloc[i]
    f2_prime_temp = df_temp['f2_prime'].iloc[i]

    # For the first curve.
    if np.abs(f1_prime_temp) == gradient1_nearly_zero_1
        or np.abs(f1_prime_temp) == gradient1_nearly_zero_2 :
            if df_temp['f1_Double_prime'].iloc[i] < 0:
                #1st CP.
                max1_critical = df_temp['x_new'].iloc[i]
            if df_temp['f1_Double_prime'].iloc[i] > 0:
                #2nd CP.
                min1_critical = df_temp['x_new'].iloc[i]

    # For the second curve.
```

```
        if np.abs(f2_prime_temp) == gradient2_nearly_zero_1
            or np.abs(f2_prime_temp) == gradient2_nearly_zero_2 :
                if df_temp['f2_Double_prime'].iloc[i] < 0:
                    #1st CP.
                    max2_critical = df_temp['x_new'].iloc[i]
                if df_temp['f2_Double_prime'].iloc[i] > 0:
                    #2nd CP.
                    min2_critical = df_temp['x_new'].iloc[i]
```

To find the full Orbital Period of the system, we need to double the value of the time interval between two critical points.

```
T_interval_1 = np.abs(max1_critical - min1_critical)
T_interval_2 = np.abs(max2_critical - min2_critical)
Avg_Time_interval = (T_interval_1 + T_interval_2)/2
Orbital_period = Avg_Time_interval*2
```

E. Determining the Mass of Stars

Stellar masses are calculated using Newtonian mechanics and the following equation from the mass function:

```
M2=(T/(2*np.pi*G.value))*(Star1_v*(Star1_v+Star2_v)**2)
M1=(Star2_v/Star1_v)*M2
M=M1+M2
```

To determine the mass of stars and thus the System's mass, we need to import $'G'$ from $astropy$ package to apply the Eq. (2).

F. Estimating the Luminosity of Binary Stars

To estimate the Luminosity of Spectroscopic binary Stars, we apply Eq.4, taking the mass of the Sun as 1.989×10^{30} Kg(expressed in program as M_0) and luminosity as 3.826×10^{26} Watts(expressed in program as L_0), respectively.

```
Lumino_1 = 1.4*L0*((M1/M0)**3.5)
Lumino_2 = 1.4*L0*((M2/M0)**3.5)
```

G. Storing the results in a destination folder

To store the outputs generated through coding, we create a folder in our working directory by writing the following code.

```
# Importing required packages
import os

# Creating a folder for storing results
folder_name = 'Binary_Star'
os.makedirs(folder_name, exist_ok=True)
base_dir = f'{folder_name}/'

#Writing onto an Excel sheet
with pd.ExcelWriter(base_dir+'New_Binary_Star.xlsx') as writer:
    df.to_excel(writer,sheet_name="DataBase_with_velocity",index=False)

#Writing onto a Text file
f = open(base_dir+"Result_of_BinaryStar.txt","w+")
f.write(f"The average velocity of Star1 is = {Star1_v:.1f} Km/sec")
f.write(f"\nThe average velocity of Star2 is = {Star2_v:.1f} Km/sec")
f.write(f"\nMass of the star1 is = {M1:.3e} Kg")
f.write(f"\nMass of the star2 is = {M2:.3e} Kg")
f.write(f"\nMass of the Binary system is = {M:.3e} Kg")
f.write(f"\nLuminosity of Star1 is = {Lumino_1:.3e} Watts")
```

```
    f.write(f"\nLuminosity of Star2 is = {Lumino_2:.3e} Watts")
    f.close()
```

Now, we are able to find all results in this folder regarding this paper.

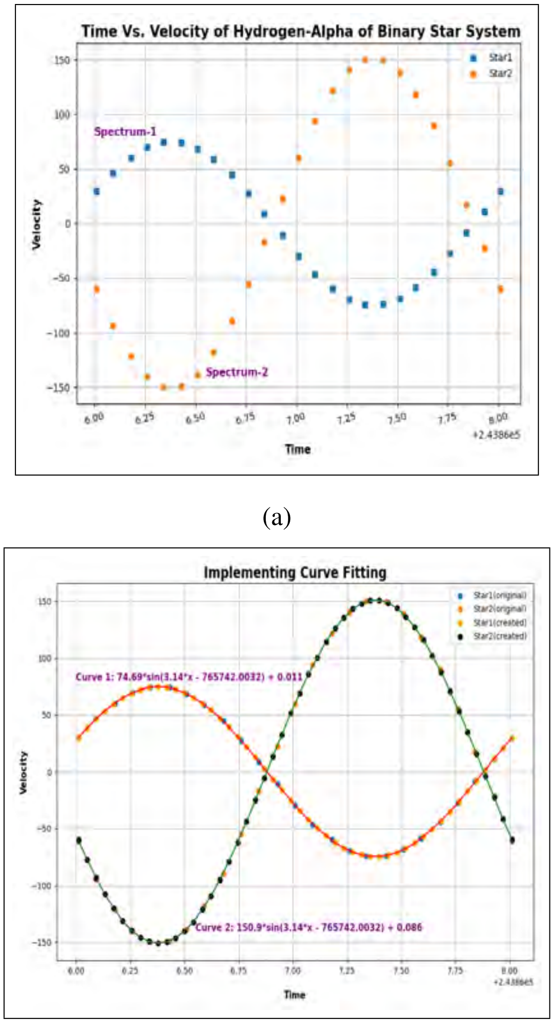
V. RESULT ANALYSIS

In Fig. 1a, we visualized the variation of the velocity for the binary star, which we obtained using doppler shift formula as given in annexure 1. Thus, fitting the curves using the Scipy-optimization package by taking the initial guess of amplitude, frequency, phase shift, and vertical shift, and minimizes the error margin(i.e., low Co-variances (here, in this case, it ranges from 0-13%)) and finding the 1st-order and 2nd-order derivatives & the critical points helps achieve the system's orbital period. The final fitted functions for those spectra are mentioned in Fig. 1b, and the findings are summarized in Table II as follows(taking the rest-frame wavelength of Hydrogen Alpha Spectrum = 6562.81 Å and the barycentric correction of Earth = 29.8 Km/sec).

From the known mass-luminosity relation(depicted in Table II) of a spectroscopic binary, we can see that the masses of the stars are 1.58 and 0.78 times the mass of the Sun respectively, and can also analyze that the massive stars are significantly more luminous; in contrast, the low-mass stars are comparatively dimmer.

Parameters	Star-1	Star-2
Velocity(V) in Km/sec	74.2	150
Mass of Spectroscopic Binaries (in Kg)	$3.139 \text{ x } 10^{30}$	$1.553 \text{ x } 10^{30}$
Luminosity (L) in Watts	$2.644 \text{ x } 10^{27}$	$2.253 \text{ x } 10^{26}$
Orbital Period (T) of Binary System in Days	2.02	2.02

TABLE II: Summary of Findings.



(a)

(b)

FIG. 1: Velocity vs time: (a) of Hydrogen-Alpha lines (b) For Curve Fitting.

VI. DISCUSSIONS

This work presents a reliable and adaptable computational method for the mass and luminosity determination of the spectroscopic binary stars. The developed algorithm successfully models radial velocity variations, estimates orbital periods, and computes stellar parameters with significant precision. By applying this methodology, we obtained masses ranging from 2 to 55 solar masses(here, in this case, it is 4.692×10^{30} Kg), subsequently deriving their respective luminosities. This tool is not only suitable for individual case studies but also scalable to handle extensive datasets from modern sky surveys. Mass determination in spectroscopic binaries relies on knowing the inclination angle(i) of the orbit, i.e., the angle between the line-of-sight and the Plane of the Binary System. In this case, it is 90^{o}; hence, $\sin(i)$ = 1.

The study enhances our ability to investigate the dynamic and evolutionary characteristics of binary stars, serving as a foundation for further developments, including eccentric orbit modelling and machine-learning integration for automatic binary classification.

VII. CONCLUSION

In future work, this algorithm can be extended to include the eccentric orbits by fitting Keplerian models, automating the detection of binary systems using machine learning, incorporating error analysis and uncertainty quantification, applying any larger and more diverse spectroscopic databases such as Gaia or SDSS, and analyzing binary populations statistically to refine mass-luminosity relations.

Appendix A

Related Formulae

All the formulas of Annexure A are obtained either from the book by S. Mahajan [12] or from the paper of O. C. Wilson [1]. So, to develop the computational code regarding the estimation of the mass and luminosity of a Spectroscopic binary, the radial velocities of each star are calculated by using the following formula(as the Earth is moving away from the celestial object [14] according to the Doppler effect [13].

$$Velocity(V) = c \cdot \left(\frac{\Delta\lambda}{\lambda_0}\right) - BC, \tag{1}$$

where, c is the speed of light in vacuum(in Km/sec), λ_0 is the rest frame wavelength of Hydrogen Alpha spectrum (expressed in Å), $\Delta\lambda$ is the difference between the rest frame wavelength and observed wavelength, BC is the barycentric correction of Earth for its motion(in Km/sec).

After calculating the mean radial velocity for both the stars and orbital period, the mass of each binary star is determined by solving the following two formulas [Eq. (2) and Eq. (3)].

$$M_1 + M_2 = \frac{T}{2\pi G} \cdot \frac{(v_1 + v_2)^3}{\sin^3 i} \tag{2}$$

$$\frac{M_1}{M_2} = \frac{v_2}{v_1} \tag{3}$$

where, M_1 and M_2 are the respective masses of stars(in Kg), $G$is the gravitational constant, T is the full orbital period(in ‘Days’, although later converted to ‘sec’ just to match with the unit of G) derived by fitting the curve(shown in Fig. 1b), v_1 and v_2 are the mean radial velocities, i is the angle between the line-of-sight and the plane of the binary system.

Now, to determine the luminosity of the stars in the system, we must use the following formula, as the mass of stars derived by solving above Eq. (2) and Eq. (3) ranges between 2 to 55 times the mass of the Sun(otherwise, we could have used another formula regarding luminosity determination).

$$L_{\mathrm{i}} = 1.4 \cdot L_0 \cdot \left(\frac{M_{\mathrm{i}}}{M_0}\right)^{3.5}, \tag{4}$$

where, M_0 and L_0 are the mass and luminosity of the Sun, respectively, L_{i} is the luminosity of each star in the system, M_{i} is the mass of each star in the system.

---

* jayashree.cs.res.2023@gmail.com

[1] O. Wilson, Astrophysical Journal **93**, 29 (1941).

[2] C. Morbey and E. Brosterhus, Publications of the Astronomical Society of the Pacific **86**, 455 (1974).

[3] S. Zucker and T. Alexander, The Astrophysical Journal **654**, L83 (2006).

[4] T. Heyne, M. Mugrauer, R. Bischoff, D. Wagner, S. Hoffmann, O. Lux, V. Munz, M. Geymeier, and R. Neuhäuser, Astronomische Nachrichten **341**, 99 (2020).

[5] M. Kounkel, K. R. Covey, K. G. Stassun, A. M. Price-Whelan, J. Holtzman, D. Chojnowski, P. Longa-Peña, C. G. Román-Zúñiga, J. Hernandez, J. Serna, *et al.*, The Astronomical Journal **162**, 184 (2021).

[6] G. Helfrich, 4 Python tools for getting started with astronomy (From opensource.com: https://opensource.com/article/19/10/python-astronomy-open-data., 2019).

[7] P. Greenfield, in Astronomical Data Analysis Software and Systems XX, Vol. 442 (2011) p. 425.

[8] B. M. Morris, E. Tollerud, B. Sipőcz, C. Deil, S. T. Douglas, J. B. Medina, K. Vyhmeister, T. R. Smith, S. Littlefair, A. M. Price-Whelan, *et al.*, The Astronomical Journal **155**, 128 (2018).

[9] Ž. Ivezić, A. J. Connolly, J. T. VanderPlas, and A. Gray, Statistics, data mining, and machine learning in astronomy: a practical Python guide for the analysis of survey data, Vol. 8 (Princeton University Press, 2020).

[10] W. Schmidt and M. Völschow, Cham, Switzerland: Springer (2021).

[11] E. Bommarito and M. Bommarito, arXiv preprint arXiv:1907.11073 (2019).

[12] S. Mahajan, Exercises in Astronomical Data Analysis for Beginners (Bhilai: OrangeBooks Publication, 2023).

[13] G. Wolfschmidt, Acta Universitatis Carolinae. Mathematica et Physica **46**, 199 (2005).

[14] M. Benacquista, An introduction to the evolution of single and binary stars, Springer Science & Business Media, (2012), p. 24.

.