Indonesian Journal of Education & Mathematical Science
Vol. 7, No. 2, Mei 2026, pp. 120~126
ISSN: 2715-985X, DOI: 10.30596/ijems.v7i2.29468

Comparative Study of Secant Method and Newton-Raphson Method in Finding the Molar Volume Value of Gas Using Python

Tatik Juwariyah[1], Damora Rhakasywi[2], Syifaul Janan[3], Asih Melati[4]
[1]Department of Industrial Engineering, Universitas Pembangunan Nasional Veteran Jakarta, Jakarta, Indonesia
[2,3]Department of MEchanical Engineering, Universitas Pembangunan Nasional Veteran Jakarta, Jakarta, Indonesia
[4]Department of Physics, Universitas Islam Negeri Sunan Kalijaga, Yogyakarta, Indonesia
[1]juwariyahtj@upnvj.ac.id, [2]rhakasywi@upnvj.ac.id, [3]syifaul.janan@upnvj.ac.id, [4]asih.melati@uin.suka.ac.id

ABSTRACT

Determining the molar volume of a real gas requires solving the non-linear Van der Waals equation. This study aims to compare the efficiency and accuracy between the Newton-Raphson Method and the Secant Method in solving the Van der Waals equation using the Python programming language. The case study examined is finding the molar volume of methane gas at a pressure of 56 atm and a temperature of 450 K. The comparison parameters used are the number of iterations and the speed of convergence with an error tolerance of 10^{-7}. The results show that the Newton-Raphson Method requires fewer iterations, namely 4 iterations with a computational time of 0.00012 s. Meanwhile, the Secant Method requires more iterations, namely 6 iterations and a computational time of 0.00015 s. The Newton-Raphson Method is less practical because it requires the calculation of the derivative of the function, while the Secant Method is more practical to be applied to Python scripts because it only requires the calculation of the Van der Waals equation function.

Keywords: Methane Gas, Molar Volume, Van der Waals, Secant, Newton Raphson

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.

Corresponding Author:
Tatik Juwariyah,
Department of Industrial Engineering,
Universitas Pembangunan Nasional "Veteran" Jakarta,
Jl. Limo Raya, Limo, 16515, Indonesia.
juwariyah_tj@upnvj.ac.id

1. INTRODUCTION

In mathematics learning, numerical modeling methods are one of the most effective tools to help students solve mathematical models in applied problems in engineering, physics, and chemistry (Kongprawechnon, 2018), (Putawa, 2023). Numerical methods are not a substitute for analytical methods, but an essential complement (Kiusalaas, 2015), (Nur Alisa, 2025). Analytical methods provide theoretical foundations, precision, and structural understanding, while numerical methods equip students with practical tools for real-world problems and develop computational thinking (Moheuddin, 2019), (Wajib Pandia, 2021). The combination of the two in the curriculum creates a more complete, relevant, and application-ready understanding of mathematics (Rika Nadila, 2022), (Azure, 2019).

In thermodynamic studies, for example, the ideal gas physical model is often not accurate enough to describe the behavior of gases at high pressures or low temperatures (Smith, 2017). In an ideal gas, particles are considered small points with no volume, allowing them to move freely in space. Meanwhile, real gases have volume, although quite small, but this volume is related to the degrees of freedom of movement in space. The ideal gas model assumes that there are no attractive or repulsive forces between particles. Particles interact only during perfectly elastic collisions. In real gases, there are attractive forces resulting from chemical bonds between atoms. At low temperatures, for example, kinetic energy decreases, particles move slowly, and attractive forces (Van der Waals or London dispersion) appear. Therefore, the limitations of the ideal gas model are corrected through the Van der Waals equation to correct for intermolecular interaction forces and obtain the molar volume parameter expressed by equation (1) (Wibowo, 2022).

$$\left(P + \frac{a}{V^2}\right)(V - b) = RT \quad (1)$$

where P: pressure (atm)
V: volume (L)
T: temperature (K)
R: gas constant (J/mol.K)
a, b: parameters according to real-world conditions (experiment)

Indonesian Journal of Education & Mathematical Science
Vol. 7, No. 2, Mei 2026, pp. 120~126
ISSN: 2715-985X, DOI: 10.30596/ijems.v7i2.29468

Mathematically, equation (1) is a third-degree polynomial with respect to volume V. Finding the roots of polynomial or non-linear equations analytically is often quite complex, so numerical methods are a practical solution (Sutrisno, 2023). Two effective and simple numerical methods are the Newton-Raphson method and the Secant method (Chapra, 2021), (Sunandar, 2020).

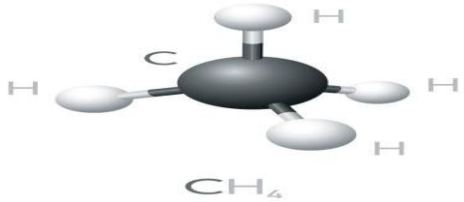


Fig. 1. Methane gas molecules (Wibowo, 2022)

This study examines the performance of the Newton-Raphson method and the Secant method in the Python programming environment to provide recommendations on which method is more optimal in obtaining the root value of a polynomial function (Nwry, 2021), especially in solving the Van der Waals equation in the case of finding the molar volume of methane gas as depicted in Figure 1.

2. RESEARCH METHOD

This study uses a numerical computational approach by comparing two numerical methods. The two methods compared are the Secant Method and the Newton-Raphson Method. In this study, the indicators of the numerical methods that are compared are: the number of iterations, iteration time and convergence speed to reach the root value of the equation. These two methods are used to obtain the polynomial root values of equation (1). The method that is able to obtain the root value with the fastest iterations or the fewest steps is considered the best method , (M., 2023), (Wibowo, 2022), (Khoiriyah, 2021). The convergence tolerance used in this study is 10-7. This means that when the difference between the next root value and the previous root value reaches 10-7, the iteration is stopped and the root value has been found.

2.1 Model Persamaan Polinomial

The Van der Waals equation is rearranged into a function *f(V)* which is in the form of a polynomial function, expressed by equation (2),

$$f(V) = PV^3 - (bP + RT)V^2 + aV - ab = 0. \quad (2)$$

2.2 Numerical Algorithms

1. *Newton-Raphson Method*: Requires one initial guess value *(V_0)* and the first derivative of the function *f'(V).* The iteration formula of the Newton-Raphson Method is stated by equation (3),

$$V_{n+1} = V_n - \frac{f(V_n)}{f'(V_n)}. \quad (3)$$

With *f(V)* expressed by equation (2), the analytical derivative of the function is expressed by equation (4).

$$f'(V) = 3PV^2 - 2(bP + RT)V + a \quad (4)$$

2.*Secant Method*: Requires two initial guess values (V_0, V_1) without needing to find the function derivative. The Secant Method formula is stated by equation (5),

$$V_{n+1} = V_n - f(V_n)\frac{V_n - V_{n-1}}{f(V_n) - f(V_{n-1})} \quad (5)$$

The flowchart of the algorithm for finding the molecular volume value or the root value of a polynomial is described in Figure 2.

Indonesian Journal of Education & Mathematical Science
Vol. 7, No. 2, Mei 2026, pp. 120~126
ISSN: 2715-985X, DOI: 10.30596/ijems.v7i2.29468



Fig. 2. Algorithm of the Newton-Raphson method and the Secant method

The Newton-Raphson method (Sechenov P.A., 2022):

1. Initialization: Enter the constants *P, T, R*, a, b, the initial guess V_0, and the error/ tolerance ε.
2. Function definition: Write the function/mathematical model *f(V)* and the derivative of the function *f'(V)*.
3. Newton Raphson Iteration: Calculate the volume value: $V_{new} = V_{old} - [f(V_{old})/f'(V_{old})]$.
4. Convergence: Check whether $|V_{new} - V_{old}|$ < error/tolerance ε.
   If Yes, then display the result V_{new} as molar volume. If No, then update $V_{old} = V_{new}$ and repeat step 3.

The Secant method (Sechenov P.A., 2022):

1. Initialization: Enter the constants *P, T, R*, a, b, the initial guess V_0 and V_1, and the error/ tolerance ε.
2. Function definition: Write the function/mathematical model *f(V)*.
3. Secant Iteration: Calculate the volume value: $V_{new} = V_2$ using the slope of the line between $(V_0, f(V_0))$ and $(V_1, f(V_1))$.
4. Convergence: Check whether $|V_2 - V_1|$ < error/tolerance ε.
   If Yes, then display the result $V_{new} = V_2$ as Molar Volume. If No, then update $V_0 = V_1$ and $V_1 = V_2$ and repeat step 3.

2.3 Implementation of Algorithms with Python Programming

Language: Python 3.11 equipped with libraries: NumPy (for computation) and Matplotlib (for visualization)
Case Study: Ammonia (NH_4) gas at P = 56 atm and T = 450 K.
The value of a = 4.17 L2.atm/mol2. The value of a represents the magnitude of the attractive forces between gas molecules.

Indonesian Journal of Education & Mathematical Science
Vol. 7, No. 2, Mei 2026, pp. 120~126
ISSN: 2715-985X, DOI: 10.30596/ijems.v7i2.29468

3. RESULTS AND DISCUSSION

A snippet of the Python programming script to solve the Van der Waals equation case study to find the molar volume of methane gas is depicted in Figure 3.

```
a = 4.17
b = 0.0371
R = 0.08206  # Konstanta gas ideal (L.atm/mol.K)
P = 56       # Tekanan (atm)
T = 450      # Suhu (K)

# 1. Definisi Fungsi f(V) dari Persamaan Van der Waals
# f(V) = PV^3 - (Pb + RT)V^2 + aV - ab = 0
def f(V):
    return P * V**3 - (P * b + R * T) * V**2 + a * V - a * b

# 2. Definisi Turunan Fungsi f'(V) untuk Newton-Raphson
def df(V):
    return 3 * P * V**2 - 2 * (P * b + R * T) * V + a

# 3. Implementasi Metode Newton-Raphson
def newton_raphson(v0, tol, max_iter):
    history = []
    v = v0
    for i in range(max_iter):
        fv = f(v)
        dfv = df(v)
        if dfv == 0: break
        v_next = v - fv / dfv
        history.append(v_next)
        if abs(v_next - v) < tol:
            return v_next, i + 1, history
        v = v_next
    return v, max_iter, history

# 4. Implementasi Metode Secant
def secant(v0, v1, tol, max_iter):
    history = []
    v_prev = v0
    v_curr = v1
    for i in range(max_iter):
        fv_curr = f(v_curr)
        fv_prev = f(v_prev)
        if (fv_curr - fv_prev) == 0: break
        v_next = v_curr - fv_curr * (v_curr - v_prev) / (fv_curr - fv_prev)
        history.append(v_next)
        if abs(v_next - v_curr) < tol:
            return v_next, i + 1, history
        v_prev = v_curr
        v_curr = v_next
    return v_curr, max_iter, history

# --- Parameter Pengujian ---
tol = 1e-7
max_iter = 100
v0_nr = 0.5          # Tebakan awal NR
v0_sec, v1_sec = 0.4, 0.5  # Tebakan awal Secant
```

Fig. 3. Python script implementing the Newton Raphson method and the Secant Method

Based on the computations carried out with a tolerance of $\varepsilon = 10^{-7}$, the data obtained are as in Table 1.

Table 1. Calculation Results of the Secant Method and the Newton-Raphson Method

Comparison Parameters	Secant Method	Newton Raphson Method
Initial Guess	V_0=0,4 and V_1=0,5	V_0=0,5
Number of Iteration	6	4
Root (Molar Volume)	0.5755 ± 10^{-7} L/mol	0.5755 ± 10^{-7} L/mol
Execution Time (average)	0.00015 s	0.00012 s

Based on Table 1, the comparative results show that the Newton-Raphson method has better computational efficiency than the Secant method in determining the molar volume of gas, even though

Indonesian Journal of Education & Mathematical Science
Vol. 7, No. 2, Mei 2026, pp. 120~126
ISSN: 2715-985X, DOI: 10.30596/ijems.v7i2.29468

both produce identical root values of 0.5755±10-7 L/mol. This advantage is reflected in the fewer number of Newton-Raphson iterations, which are achieved in the fourth iteration while with the Secant method the root value is obtained in the sixth iteration. This has a direct impact on the shorter average execution time (0.00003 s) the time difference between the Newton-Raphson method and the Secant method. In addition, in terms of input parameters, Newton-Raphson is proven to be more practical because it only requires one initial guess (V0=0.5), while the Secant method requires two initial guesses (V0=0.4 and V1=0.5), so that from the number of iterations and iteration time indicators the Newton-Raphson method provides more optimal performance to achieve convergence in this calculation.

The convergence rate of the iteration process of both methods is presented in Figure 4. The mathematical function of the Van der Waals equation and the molar volume value are analytically presented in Figure 5.

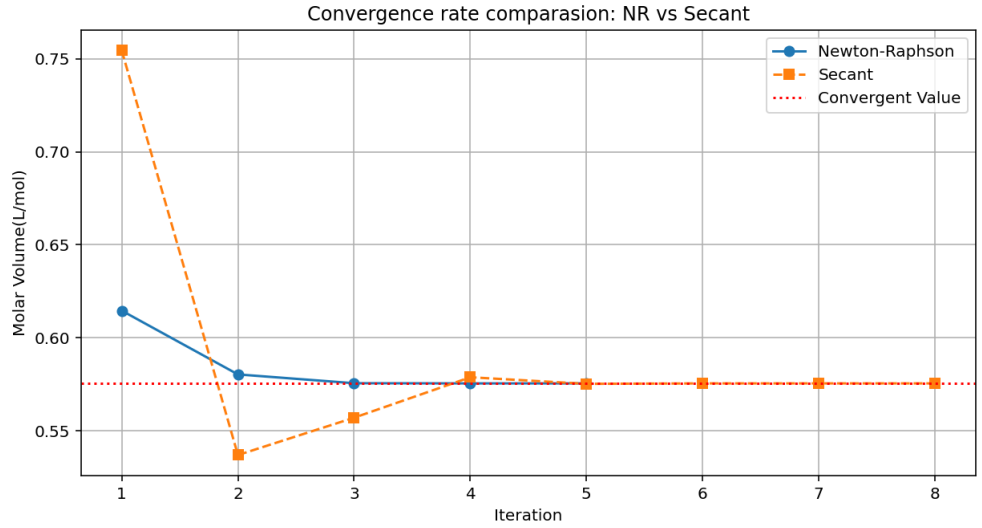


Fig. 4. Convergence rate of the Secant method and the Newton Raphson method

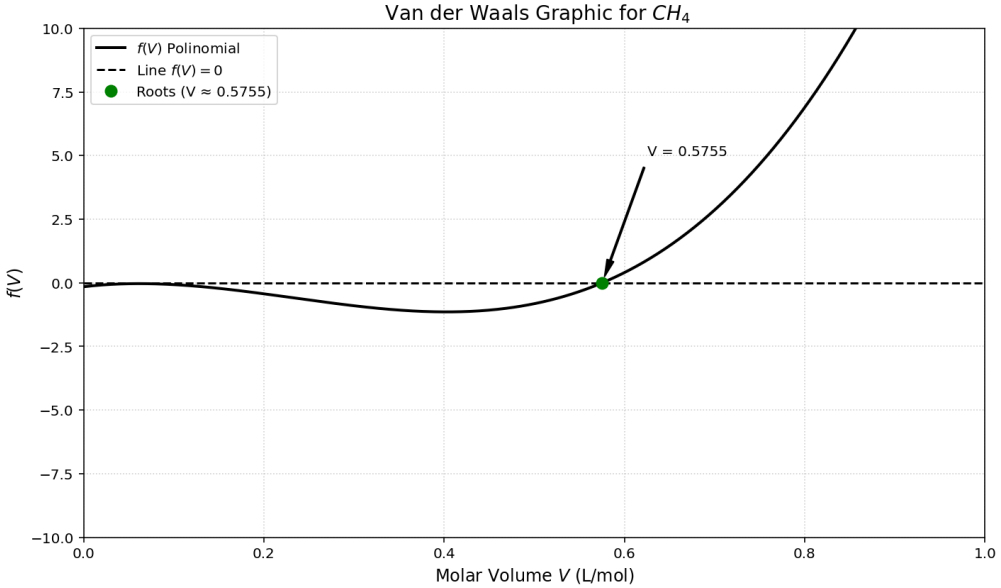


Fig. 5. Graph of the Van der Waals equation and the root value of the function

A comparative analysis of the Secant Method and the Newton Raphson Method in terms of convergence, ease of implementation in the Python programming language, efficiency and accuracy of numerical solutions is presented in Table 2.

Indonesian Journal of Education & Mathematical Science
Vol. 7, No. 2, Mei 2026, pp. 120~126
ISSN: 2715-985X, DOI: 10.30596/ijems.v7i2.29468

Table 2. Comparison of the Secant Method and the Newton Raphson Method

	Secant Method	Newton Raphson Method
Convergence	Convergence is slower than the Newton-Raphson method, as seen from the greater number of iterations (6 iterations). However, this method has the advantage that it only requires the definition of a mathematical function f(V) that is continuous on the interval V_0 and V_1. In cases where the first derivative of a mathematical function is difficult to determine analytically, the Secant method is recommended for use because it does not require information about the definition of the first derivative of the mathematical function.	Shows quadratic convergence, as seen from the smaller number of iterations (4 iterations). In cases that require long iteration steps, the Newton Raphson method is recommended because it has implications for the computer memory required. However, this method has a weakness if the derivative f'(V) is difficult to determine or is close to zero.
Ease of Implementation in Python Language	Excellent, efficient and practical in writing Python scripts considering that it is enough to write the definition of mathematical functions. *f(V)* (Kiusalaas, 2015)	Excellent, efficient, but less practical if the mathematical derivative function f'(V) involves certain Python library functions
Efficiency	Time to get the solution of the root value of the function or molar volume = 0.00015 s. In large computations, it is less efficient in terms of computation time to achieve numerical convergence.	The time to obtain the solution for the root value of the function or molar volume = 0.00012 s. In large computations, it is very efficient in terms of computation time to achieve numerical convergence (Osniman Paulina Maure, 2024).
Accuracy	Accurate, in the case of finding the molar volume value of CH_4, the value obtained is close to the analytical value, namely V = 0.5755 L/mol with a tolerance of 10^{-7}	Accurate, in the case of finding the molar volume value of CH_4, the value obtained is close to the analytical value, namely V = 0.5755 L/mol with a tolerance of 10^{-7}

4. CONCLUSION

Both methods successfully found the molar volume of methane gas with identical accuracy. The Newton-Raphson method is the best method if speed is the main priority and the derivative function can be easily defined. In the case of finding the molar volume of CH4 gas, at a pressure of 56 atm and a temperature of 450 K, with an error tolerance of 10^{-7} m, the Newton-Raphson method provided superior results, reaching convergence after 4 iterations with a computational time of 0.00012 s, while the Secant method reached convergence after 6 iterations with a computational time of 0.00015 s. The Secant method is more recommended practically when the derivative function is difficult to determine and write in a Python script.

REFERENCES

Azure, I. A. (2019). Comparative Study of Numerical Methods for Solving Non-linear Equations Using Manual Computation. *Mathematics Letters, 5*(4), 41-51. Retrieved from https://doi.org/10.11648/j.ml.20190504.11

Chapra, S. C. (2021). *Numerical Methods for Engineers (8th ed.).* McGraw-Hill Education.

Khoiriyah, I. (2021). *Modifikasi Metode Newton-Secant Dalam Penyelesaian Persamaan Non-linier Yang Memiliki Multiplisitas.* Malang: Fakultas Sains dan TEknologi. UIN Maulana Malik Ibrahim.

Kiusalaas, J. (2015). *Numerical Methods in Engineering with Python 3.* Cambridge University Press.

Kongprawechnon, W. &. (2018). Comparison of Numerical Methods for Solving Nonlinear Equations in Chemical Engineering. *Journal of Applied Science and Engineering*.

M., M. (2023). Modifikasi Garis Singgung Untuk Mempercepat Iterasi Pada Metode Newton Raphson. *Euler : Jurnal Ilmiah Matematika, Sains dan Teknologi, 11*(2), 351–360.

Moheuddin, M. M. (2019). A New Study To Find Out The Best Computational Method For Solving. *Applied Mathematics and Sciences An International Journal (MathSJ), 6*(3), 15–31. doi:10.5121/mathsj.2019.6302

Nur Alisa, A. I. (2025). Application of Runge Kutta Fehlberg (RKF45) Method as a Numerical Analysis to SIR Model of Tuberculosis Transmission in Central Java. *Indonesian Journal of Education & Mathematical Science (IJEMS), 6*(3), 201-208.

Nwry, A. W. (2021). Comparison Between Bisection, Newton and Secant Methods for determining the root of the Non-Linear equation using MATLAB. *Turkish Journal of Computer and Mathematics Education*(12), 1115–1122.

Osniman Paulina Maure, H. T. (2024). Studi Komparasi Beberapa Metode Numerik Dalam Mengaproksimasi Akar-Akar Persamaan Non-Linier. *ASIMTOT, 6*(2), 1-12.

Putawa, R. A. (2023). Metode Numerik dalam Perspektif Pragmatisme Dan Relevansinya dengan Bidang Keteknikan. *Jurnal Filsafat Indonesia, 6*(1), 60–65.

Rika Nadila, K. K. (2022). Deskripsi Creative Thinking pada Materi Segitiga dan Segi Empat ditinjau dari Stres Akademik Siswa. *VENN: Journal of Sustainable Innovation on Education, Mathematics and Natural Sciences*.

Sechenov P.A., R. I. (2022). Numerical method and mathematical model for finding the equilibrium composition of the thermodynamic system of the T-Energy software package. *Herald of Dagestan State Technical University. Technical Sciences, 49*(4), 104-112.

Smith, J. M. (2017). *Introduction to Chemical Engineering Thermodynamics.* McGraw-Hill Education.

Sunandar, E. &. (2020). Perbandingan Metode Newton-Raphson & Metode Secant Untuk Mencari Akar Persamaan Dalam Sistem Persamaan Non-Linier. *Petir : Jurnal Pengkajian Dan Penerapan Teknik Informatika, 13*(1), 72-79. Retrieved from https://doi.org/10.33322/petir.v13i1.893

Sutrisno, T. (2023). Aplikasi Penyelesaian Numerik Pencarian Akar Persamaan Non-Linier Dan Penerapannya Dalam Menyelesaikan Analisis Break Even Point. *Computatio: Journal of Computer Science and Information Systems, 7*(1), 37–49.

Wajib Pandia, I. S. (2021). Penentuan Akar Persamaan Nonlinier Dengan Metode Numerik. *Jurnal Mutiara Pendidikan, 6*(2), 122-129.

Wibowo, A. (2022). Implementasi Python dalam Penyelesaian Masalah Termodinamika Gas Nyata. *Jurnal Komputasi Kimia Indonesia, 5*(2).