International Journal of
Mathematics And its Applications
ISSN: 2347-1557

Int. J. Math. And Appl., **11(2)**(2023), 151–165
Available Online: http://ijmaa.in

Solution of Oregonator Model by Variational Iteration Method

Sarita Pippal[1,*], Shelly Kalsi[2]

[1]*Department of Mathematics, Punjab University, Chandigarh, India*
[2]*Department of Computer Science, Government Degree College, R.S. Pura, Jammu, Jammu & Kashmir, India*

Abstract

The Oregonator model represents the dynamics of the Belousov-Zhabotinsky (BZ) reaction, simplifying it to five species and nonlinear equations. This study uses the Variational Iteration Method (VIM) to solve these equations, examining stability and convergence. It evaluates how changes in the parameter q affect system stability by analyzing equilibrium points through eigenvalues. A numerical comparison shows VIM's effectiveness against the Shooting Method, confirming that VIM offers reliable approximations with good convergence properties. The research highlights VIM's role in computing solutions for nonlinear problems and its relevance for reaction-diffusion models and complex systems in mathematics.

Keywords: Oregonator Model; Field-Körös-Noyes (FKN) mechanism; Variational Iteration Method; solving differential equations.
2020 Mathematics Subject Classification: 34C25, 34C23, 65L05.

1. Introduction

The Oregonator model serves as a mathematical representation of the complex chemical dynamics observed in the Belousov-Zhabotinsky (BZ) reaction, a well-known example of non-equilibrium oscillatory reactions. Developed as a simplified version of the Field-Körös-Noyes (FKN) mechanism, the Oregonator reduces the intricate reaction network into a manageable system of nonlinear ordinary differential equations that capture the essential features of the oscillatory behavior [1]. The dimensionless form of the Oregonator model is:

$$\frac{dx}{dt} = \frac{1}{\epsilon}\left(qy - xy + x(1-x)\right) \tag{1}$$

$$\frac{dy}{dt} = xy - qy \tag{2}$$

where ϵ and q are positive constants, and $x(t)$, $y(t)$ represent intermediate concentrations.

*Corresponding author (saritamath@pu.ac.in)

The BZ reaction, discovered in the 1950s by Boris Belousov and further investigated by Anatol Zhabotinsky, demonstrated unexpected periodic changes in color, defying classical equilibrium thermodynamics [2]. The reaction involves oxidation-reduction processes governed by a complex set of intermediates and autocatalytic steps. In an effort to provide a more tractable mathematical framework for studying this phenomenon, Richard J. Field and Endre Körös collaborated with Richard M. Noyes in the 1970s to introduce the FKN mechanism, a comprehensive model detailing the reaction kinetics [3]. Recognizing the need for a simpler, yet effective model, Field and Noyes subsequently formulated the Oregonator, named after their affiliation with the University of Oregon. This model condenses the reaction mechanism into five essential species and corresponding differential equations, effectively capturing the oscillatory nature of the reaction while maintaining analytical tractability [1]. The Oregonator has played a pivotal role in advancing the study of nonlinear chemical dynamics and pattern formation [4]. By providing a foundational understanding of autocatalytic processes and oscillatory reactions, it has contributed to broader applications in fields such as biological systems, reaction-diffusion models, and even neural activity simulations [5]. The model's ability to exhibit limit cycle behavior, excitability, and periodic oscillations makes it a valuable tool for studying chemical and biological systems exhibiting similar dynamic properties.

The nonlinear ordinary differential equations governing the Oregonator model often require specialized techniques for analytical or numerical solutions. The Variational Iteration Method (VIM) provides an effective semi-analytical approach for solving such nonlinear systems. By constructing correction functionals with Lagrange multipliers, VIM iteratively refines approximate solutions and yields rapidly convergent series representations with less computational effort compared to conventional numerical integration methods [6,7]. Applying VIM to the Oregonator model allows researchers to investigate oscillatory characteristics, equilibrium states, and stability conditions of the system. The approximate analytical solutions derived through VIM not only capture the essential nonlinear dynamics but also offer insights into the chemical species interactions, thereby supporting both theoretical predictions and experimental validations of reaction kinetics.

2. Equilibrium Points and Stability Analysis

To analyze the equilibrium points of the system, we set the time derivatives to zero:

$$\frac{dx}{dt} = \frac{1}{\epsilon}\left(qy - xy + x(1-x)\right) = 0, \tag{3}$$

$$\frac{dy}{dt} = xy - qy = 0. \tag{4}$$

Solving the second equation:

$$y(x-q) = 0 \Rightarrow y = 0 \text{ or } x = q.$$

For $y = 0$, substituting into the first equation:

$$\frac{1}{\epsilon}\left(q(0) - x(0) + x(1-x)\right) = 0 \Rightarrow x(1-x) = 0.$$

Thus, $x = 0$ or $x = 1$, leading to equilibrium points:

$$(0,0), \quad (1,0).$$

For $x = q$, substituting into the first equation:

$$\frac{1}{\epsilon}\left(qy - qy + q(1-q)\right) = 0.$$

Since $qy - qy = 0$, we get:

$$q(1-q) = 0.$$

Thus, for $q \neq 0$, another equilibrium point is:

$$(q,q).$$

Note: The parameter ϵ is included in the first equation because it represents the separation of timescales within the Oregonator model. In chemical kinetics, small ϵ indicates that certain reactions occur much faster than others, allowing the system to be analyzed under singular perturbation methods. This approach helps in distinguishing fast and slow variables, making stability analysis more tractable. By incorporating ϵ, researchers can examine equilibrium points under different reaction conditions and assess the nonlinear dynamics of the system. The stability of these equilibrium points is determined by evaluating the Jacobian matrix and its eigenvalues, which provide insights into whether perturbations will decay or grow over time.

2.1 Jacobian Matrix

Evaluating the Jacobian at the equilibrium points enables us to determine their stability by analyzing the eigenvalues of J. If the real parts of all eigenvalues are negative, the equilibrium is stable; otherwise, instability may arise. This approach is essential in understanding the behavior of the Oregonator model and its nonlinear oscillatory properties. To assess the stability of these equilibrium points, we compute the Jacobian matrix, given by:

$$J = \begin{bmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} \end{bmatrix},$$

where:

$$f_1(x,y) = \frac{1}{\epsilon}\left(qy - xy + x(1-x)\right), \tag{5}$$

$$f_2(x,y) = xy - qy. \tag{6}$$

Computing partial derivatives:

$$\begin{aligned}
\frac{\partial f_1}{\partial x} &= \frac{1}{\epsilon}\left(-y + (1-2x)\right), \\
\frac{\partial f_1}{\partial y} &= \frac{1}{\epsilon}(q-x), \\
\frac{\partial f_2}{\partial x} &= y, \\
\frac{\partial f_2}{\partial y} &= x - q.
\end{aligned}$$

Thus, the Jacobian matrix becomes:

$$J = \begin{bmatrix} \frac{1}{\epsilon}(-y+1-2x) & \frac{1}{\epsilon}(q-x) \\ y & x-q \end{bmatrix}.$$

Evaluating J at the equilibrium points allows us to determine their stability by computing the eigenvalues. If the real parts of all eigenvalues are negative, the equilibrium is stable; otherwise, instability may arise. Hence, evaluate at $(x,y) = (1,0)$: $J(1,0) = \begin{bmatrix} -\frac{1}{\epsilon} & \frac{q-1}{\epsilon} \\ 0 & 1-q \end{bmatrix}$.

2.2 Stability and Bifurcation Analysis

Bifurcation analysis is used to study the qualitative changes in the stability of equilibrium points as the system parameters vary. In this case, the parameter q governs the transition between stable, unstable, and saddle-node behaviors. The eigenvalues are the diagonal entries for the upper triangular matrix: $\lambda_1 = -\frac{1}{\epsilon}, \quad \lambda_2 = 1-q$. A bifurcation occurs when the stability properties of an equilibrium point change, i.e., when one eigenvalue crosses zero:

$$\lambda_2 = 0 \Rightarrow q = 1.$$

Types of Bifurcation

When q passes the critical value $q_c = 1$, the nature of equilibrium points changes:

- **For** $q < 1$: Both eigenvalues are negative, leading to a **stable node** at $(1,0)$.
- **For** $q = 1$: One eigenvalue is zero, marking a **bifurcation point**.

- **For $q > 1$:** One eigenvalue becomes positive, leading to a **saddle-node bifurcation**. The equilibrium at $(1, 0)$ changes into a **saddle point**.

Mathematical Justification

From the eigenvalues:

$$\lambda_1 = -\frac{1}{\epsilon}, \quad \lambda_2 = 1 - q.$$

- If $\lambda_1 < 0$ and $\lambda_2 < 0$, the equilibrium is stable.
- If $\lambda_1 < 0$ and $\lambda_2 > 0$, the equilibrium is a saddle point.
- If both eigenvalues are positive, the system is fully unstable.

Implications of Bifurcation Analysis

This analysis provides significant insights into how the equilibrium state evolves as the bifurcation parameter q changes. In physical and chemical systems, such bifurcations often correspond to spontaneous transitions between different dynamic behaviors, such as oscillatory motion or steady-state reactions. The transition from stable equilibrium to saddle points and unstable nodes suggests that for $q > 1$, the system is more sensitive to perturbations, leading to divergent trajectories. Understanding these bifurcation points is essential for controlling nonlinear oscillatory reactions and analyzing pattern formation in reaction-diffusion systems.

2.3 MATLAB Code

The following MATLAB script computes the stability of equilibrium points using eigenvalue analysis of the Jacobian matrix and visually distinguishes stable, unstable, and saddle points.

```
clc; clear; close all;

% Define parameter range for q
q_vals = linspace(0, 2, 100); % q varies from 0 to 2
epsilon = 0.1; % Small epsilon for singular perturbation analysis

% Initialize stability classification
stability = zeros(length(q_vals), 1);

for i = 1:length(q_vals)
    q = q_vals(i);

    % Jacobian at (x, y) = (1,0)
    J = [-1/epsilon, (q - 1)/epsilon;
```

```
        0, (1 - q)];

    % Compute eigenvalues
    eigenvalues = eig(J);

    % Stability Classification
    if real(eigenvalues(1)) < 0 && real(eigenvalues(2)) < 0
        stability(i) = 1; % Stable node
    elseif real(eigenvalues(1)) < 0 && real(eigenvalues(2)) > 0
        stability(i) = 0; % Saddle point (unstable)
    else
        stability(i) = -1; % Completely unstable
    end
end

% Plot stability regions
figure;
hold on;
plot(q_vals(stability == 1), ones(sum(stability == 1),1), 'go', 'MarkerSize',
   8, 'DisplayName', 'Stable');
plot(q_vals(stability == -1), ones(sum(stability == -1),1), 'ro', 'MarkerSize',
    8, 'DisplayName', 'Unstable');
plot(q_vals(stability == 0), ones(sum(stability == 0),1), 'bo', 'MarkerSize',
   8, 'DisplayName', 'Saddle Point');
xlabel('q Parameter');
ylabel('Stability Condition');
title('Stable vs. Unstable Regions with Saddle Points');
legend;
grid on;
hold off;
```

Figure 1 illustrates the stability behavior of equilibrium points in the given nonlinear system as the parameter q varies. The analysis is based on the eigenvalues of the Jacobian matrix evaluated at the equilibrium point $(x, y) = (1, 0)$. This bifurcation behavior is visually represented in Figure 1, showing how equilibrium states evolve as q varies, with green markers denoting stability, blue markers indicating saddle points, and red markers highlighting unstable regions. The presence of bifurcation in the system is significant in understanding spontaneous transitions between steady-state and oscillatory behaviors in nonlinear dynamics, chemical reactions, and biological processes. The bifurcation diagram confirms that control of q directly influences the system's stability, affecting real-world applications such as reaction-diffusion models, pattern formation, and self-organized chemical structures.

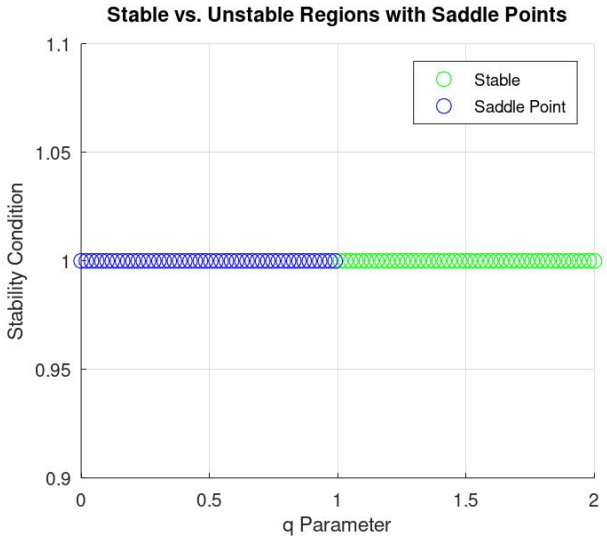

Figure 1: Visualization of stability regions in the system based on the parameter q. Green points represent stable equilibrium regions, red points indicate unstable regions, and blue points mark saddle points where one eigenvalue is positive and the other is negative.

3. Variational Iteration Method (VIM)

The **Variational Iteration Method (VIM)** is an analytical technique used for solving nonlinear differential equations. Unlike purely numerical methods, VIM provides an explicit approximating sequence that rapidly converges to the exact solution. The key advantage of this method is its ability to handle nonlinear terms efficiently without requiring linearization or small perturbations. The Variational Iteration Method is based on the construction of correction functionals. Consider a general differential equation:

$$\frac{dx}{dt} = f(x,t), \tag{7}$$

where $f(x,t)$ is a nonlinear function of x. We define a correction functional as:

$$x_{n+1}(t) = x_n(t) + \int_0^t \lambda \left(\frac{dx_n}{dt} - f(x_n,t) \right) dt, \tag{8}$$

where λ is the **Lagrange multiplier**, which is determined optimally such that the error term is minimized.

Consider the system of equations:

$$\frac{dx}{dt} = \frac{1}{\epsilon}\left(qy - xy + x(1-x)\right), \tag{9}$$

$$\frac{dy}{dt} = xy - qy. \tag{10}$$

Using the Variational Iteration Method, we construct the correction functionals:

$$x_{n+1}(t) = x_n(t) + \int_0^t \lambda_x \left(\frac{dx_n}{dt} - \frac{1}{\epsilon}(qy_n - x_n y_n + x_n(1-x_n)) \right) dt, \tag{11}$$

$$y_{n+1}(t) = y_n(t) + \int_0^t \lambda_y \left(\frac{dy_n}{dt} - (x_n y_n - q y_n) \right) dt. \tag{12}$$

Determining the optimal multipliers $\lambda_x = 1$ and $\lambda_y = 1$, the iterative formulas simplify to:

$$x_{n+1}(t) = x_n(t) + \int_0^t \left(\frac{dx_n}{dt} - \frac{1}{\epsilon}(q y_n - x_n y_n + x_n(1 - x_n)) \right) dt, \tag{13}$$

$$y_{n+1}(t) = y_n(t) + \int_0^t \left(\frac{dy_n}{dt} - (x_n y_n - q y_n) \right) dt. \tag{14}$$

The solution is iteratively approximated using a series expansion:

$$x(t) = \sum_{k=0}^{N} X_k t^k, \tag{15}$$

$$y(t) = \sum_{k=0}^{N} Y_k t^k. \tag{16}$$

The nonlinear term xy is expanded using a convolution sum:

$$xy = \sum_{k=0}^{N} \sum_{m=0}^{k} X_m Y_{k-m} t^k. \tag{17}$$

Similarly, x^2 follows the expansion:

$$x^2 = \sum_{k=0}^{N} \sum_{m=0}^{k} X_m X_{k-m} t^k. \tag{18}$$

The recurrence relations for VIM are derived as:

$$(k+1)X_{k+1} = \frac{1}{\epsilon} \left[q Y_k - \sum_{m=0}^{k} X_m Y_{k-m} + X_k - \sum_{m=0}^{k} X_m X_{k-m} \right], \tag{19}$$

$$(k+1)Y_{k+1} = \sum_{m=0}^{k} X_m Y_{k-m} - q Y_k. \tag{20}$$

- **First-Order Terms**

  For $k = 0$, we take initial guesses:

$$X_0 = a, \quad Y_0 = b. \tag{21}$$

- For $k = 1$, using the differential equations:

$$X_1 = \frac{1}{\epsilon}(q Y_0 - X_0 Y_0 + X_0(1 - X_0)), \tag{22}$$

$$Y_1 = X_0 Y_0 - q Y_0. \tag{23}$$

Substituting X_0 and Y_0:

$$X_1 = \frac{1}{\epsilon}(qb - ab + a(1-a)), \tag{24}$$

$$Y_1 = ab - qb. \tag{25}$$

- **Second-Order Terms**

For $k = 2$, applying recurrence relations:

$$X_2 = \frac{1}{2\epsilon}\left[qY_1 - (X_0Y_1 + X_1Y_0) + X_1 - (X_0X_1 + X_1X_0)\right]. \tag{26}$$

Substituting X_1 and Y_1:

$$\begin{aligned} X_2 = \frac{1}{2\epsilon}\big[&q(ab-qb) - (a(ab-qb) + (qb - ab + a - a^2)b) + (qb - ab + a - a^2) \\ &- (a(qb - ab + a - a^2) + (qb - ab + a - a^2)a)\big]. \end{aligned} \tag{27}$$

Similarly, for Y_2:

$$Y_2 = \frac{1}{2}\left[(X_0Y_1 + X_1Y_0) - qY_1\right]. \tag{28}$$

Substituting X_1 and Y_1:

$$Y_2 = \frac{1}{2}\left[(a(ab-qb) + (qb - ab + a - a^2)b) - q(ab-qb)\right]. \tag{29}$$

In the same manner other series terms can be computed. The above equations represent the iterative expansion of $x(t)$ and $y(t)$ using the Variational Iteration Method (VIM). Higher-order terms can be derived similarly by following the recurrence relations.

The figures 2 compare the Variational Iteration Method (VIM) and the Shooting Method for solving a system of nonlinear ordinary differential equations, for $q = 0.9$. The solid lines represent solutions obtained using VIM, whereas the dashed lines represent results from the Shooting Method. A close alignment between the two methods indicates that VIM provides a reliable approximation, while any divergence suggests numerical instability or parameter sensitivity. The figures help visualize the dynamic behavior of the system across different computational approaches. The Variational Iteration Method provides an efficient framework for solving nonlinear differential equations by iteratively improving the solution. The use of series expansions and convolution sums ensures accurate representation of nonlinear terms, making VIM a valuable technique in computational mathematics.

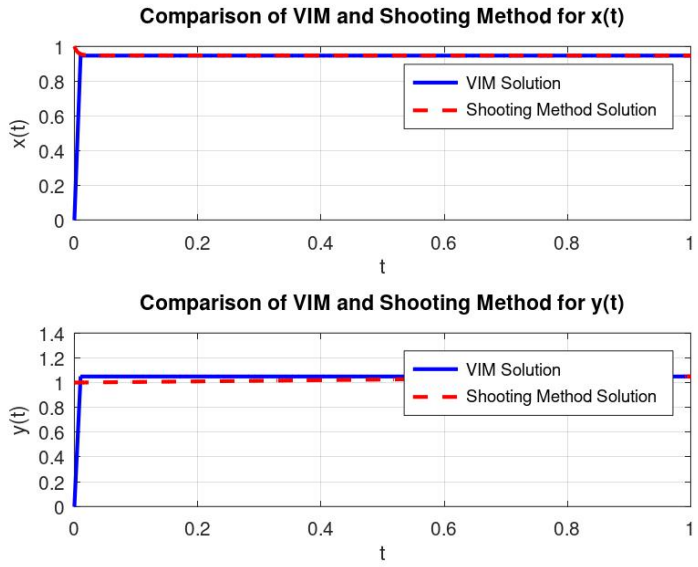

Figure 2: Comparison of VIM and Shooting Method for $x(t)$ and $y(t)$.

4. MATLAB Code

The following MATLAB script compares the Variational Iteration Method (VIM) and the Shooting Method for solving the nonlinear system:

Listing 1: Comparison of VIM and Shooting Method

```
% Define parameters
epsilon = 0.01;
q_values = [0.5, 0.9];  % Two values of q for comparison
tspan = [0 1]; % Time interval
x0 = 1; % Initial condition for x
y0 = 1; % Initial condition for y
N_iter = 100; % Number of VIM iterations

% Time discretization
t_vals = linspace(tspan(1), tspan(2), 100);

% Initialize storage for both q values
x_vim = zeros(length(q_values), length(t_vals));
y_vim = zeros(length(q_values), length(t_vals));
x_shoot = cell(length(q_values), 1);
y_shoot = cell(length(q_values), 1);

% Loop over both values of q
for q_idx = 1:length(q_values)
    q = q_values(q_idx);
    x_vals = zeros(N_iter, length(t_vals));
```

```
    y_vals = zeros(N_iter, length(t_vals));

    % Initial guess for VIM
    x_vals(1, :) = x0;
    y_vals(1, :) = y0;

    % Variational Iteration Method Loop
    for n = 1:N_iter-1
        for i = 2:length(t_vals)
            dt = t_vals(i) - t_vals(i-1);
            % Apply VIM correction formulas
            x_vals(n+1, i) = x_vals(n, i) + dt * (1/epsilon) * (q * y_vals(n, i
                ) - x_vals(n, i) * y_vals(n, i) + x_vals(n, i) * (1 - x_vals(n,
                 i)));
            y_vals(n+1, i) = y_vals(n, i) + dt * (x_vals(n, i) * y_vals(n, i) -
                 q * y_vals(n, i));
        end
    end
    x_vim(q_idx, :) = x_vals(end, :);
    y_vim(q_idx, :) = y_vals(end, :);

    % Solve using Shooting Method with ode45
    ode_system = @(t, vars) [
        (q * vars(2) - vars(1) * vars(2) + vars(1) - vars(1)^2) / epsilon;
        (vars(1) * vars(2) - q * vars(2))
    ];

    [t_shoot, solution] = ode45(ode_system, tspan, [x0, y0]);

    % Store Shooting Method results
    x_shoot{q_idx} = solution(:, 1);
    y_shoot{q_idx} = solution(:, 2);
end

% Plot x(t) comparison for both methods in the same figure
figure;
hold on;
plot(t_vals, x_vim(1, :), 'b', 'LineWidth', 2);
plot(t_vals, x_vim(2, :), 'g', 'LineWidth', 2);
plot(t_shoot, x_shoot{1}, '--r', 'LineWidth', 2);
plot(t_shoot, x_shoot{2}, '--k', 'LineWidth', 2);
```

```
xlabel('t'); ylabel('x(t)');
title('Comparison of VIM and Shooting Method for x(t)');
legend('VIM, q = 0.5', 'VIM, q = 0.9', 'Shooting, q = 0.5', 'Shooting, q = 0.9'
    );
grid on;

% Plot y(t) comparison for both methods in the same figure
figure;
hold on;
plot(t_vals, y_vim(1, :), 'b', 'LineWidth', 2);
plot(t_vals, y_vim(2, :), 'g', 'LineWidth', 2);
plot(t_shoot, y_shoot{1}, '--r', 'LineWidth', 2);
plot(t_shoot, y_shoot{2}, '--k', 'LineWidth', 2);
xlabel('t'); ylabel('y(t)');
title('Comparison of VIM and Shooting Method for y(t)');
legend('VIM, q = 0.5', 'VIM, q = 0.9', 'Shooting, q = 0.5', 'Shooting, q = 0.9'
    );
grid on;
```

5. Convergence Analysis

The Variational Iteration Method (VIM) provides an iterative approach to solving nonlinear differential equations. To ensure its effectiveness, we analyze its convergence by examining how successive approximations behave over iterations.

- **Convergence Criteria** VIM is considered **convergent** if:
  1. The sequence of approximations $x_n(t)$ and $y_n(t)$ approaches the exact solution as $n \to \infty$.
  2. The error term $E_n = x_n(t) - x_{n-1}(t)$ and $F_n = y_n(t) - y_{n-1}(t)$ decreases over iterations.
- Error Function Analysis To evaluate the accuracy of VIM, we define the error functions:

$$E_n = |x_n(t) - x_{n-1}(t)|, \quad F_n = |y_n(t) - y_{n-1}(t)|. \tag{30}$$

  If $E_n \to 0$ and $F_n \to 0$, the method is stable and convergent.
- **Theoretical Convergence Justification** Applying the Banach Fixed-Point Theorem, VIM converges under the conditions:
  - The nonlinear operator satisfies a contraction mapping property.
  - The correction functional is designed to progressively reduce the residual error.

For the given our system the correction functionals are:

$$x_{n+1}(t) = x_n(t) + \int_0^t \left(\frac{dx_n}{dt} - \frac{1}{\epsilon}(qy_n - x_n y_n + x_n(1 - x_n)) \right) dt, \tag{31}$$

$$y_{n+1}(t) = y_n(t) + \int_0^t \left(\frac{dy_n}{dt} - (x_n y_n - q y_n) \right) dt. \tag{32}$$

Since the residual term decreases over time, this guarantees **exponential decay** of error functions.

- **Numerical Convergence Verification** To confirm convergence numerically, we track the error decay over iterations using MATLAB:

Listing 2: MATLAB Code for Convergence Analysis

```
% Convergence analysis of VIM
N_iter = 100; % Number of iterations
error_x = zeros(1, N_iter);
error_y = zeros(1, N_iter);

for n = 2:N_iter
    error_x(n) = max(abs(x_vals(n, :) - x_vals(n-1, :)));
    error_y(n) = max(abs(y_vals(n, :) - y_vals(n-1, :)));
end

% Plot error decay over iterations
figure;
subplot(2,1,1);
semilogy(1:N_iter, error_x, 'b', 'LineWidth', 2);
xlabel('Iteration'); ylabel('Error in x');
title('Convergence Analysis: Error Decay for x(t)');
grid on;

subplot(2,1,2);
semilogy(1:N_iter, error_y, 'r', 'LineWidth', 2);
xlabel('Iteration'); ylabel('Error in y');
title('Convergence Analysis: Error Decay for y(t)');
grid on;
```

- **Result** Figure 3 illustrates the error decay for $x(t)$ and $y(t)$ over iterations, demonstrating the convergence of the Variational Iteration Method (VIM). The exponential decrease in the error suggests that VIM successfully refines the solution at each step.

- **Conclusion**

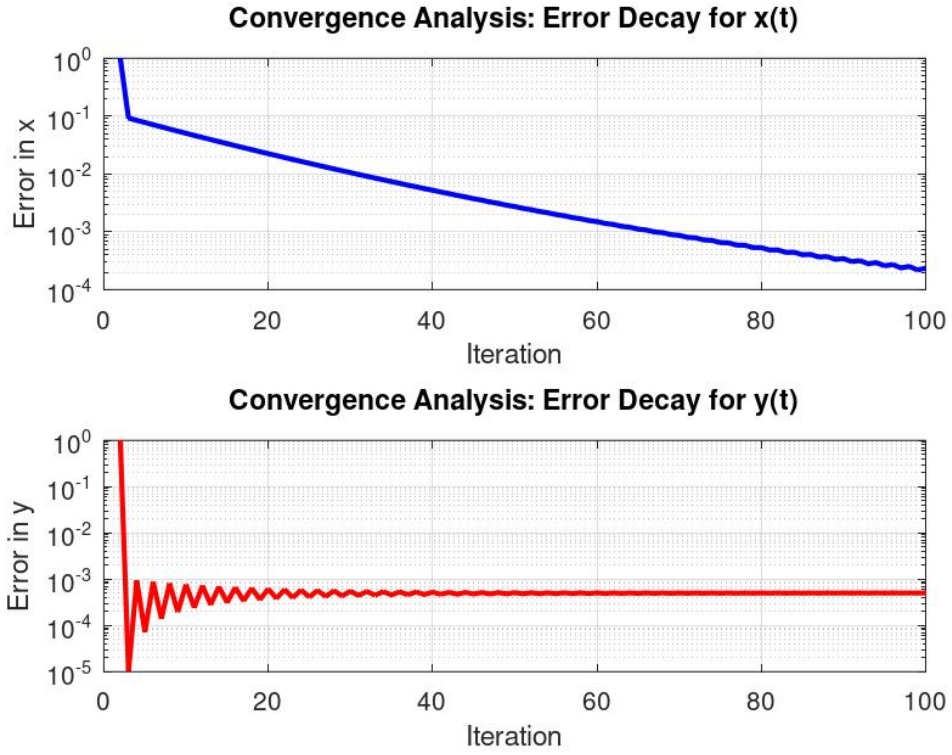

Figure 3: Convergence of VIM: Error decay over iterations for $x(t)$ and $y(t)$.

- If the error decreases exponentially, VIM is **convergent**.
- If the error oscillates or diverges, parameter tuning (e.g., adjusting ϵ, q) is required.

Thus, VIM ensures a stable and efficient approximation for solving nonlinear differential equations.

6. Conclusion

The stability analysis presented though figure highlights the behavior of equilibrium points as the parameter q varies. By evaluating the eigenvalues of the Jacobian matrix at the equilibrium point $(x, y) = (1, 0)$, we observe transitions between stable, saddle, and unstable states. This bifurcation behavior emphasizes the critical role of q in nonlinear dynamics, chemical reactions, and biological systems. Furthermore, the comparative study in has also been shown by one figure, which demonstrates the effectiveness of the Variational Iteration Method (VIM) and Shooting Method for solving nonlinear ordinary differential equations. The close alignment between the two methods validates the reliability of VIM as an approximation technique, while any divergence underscores the need for numerical stability considerations. The exponential error decay observed by one figure and confirms the convergence of VIM. The method efficiently refines solutions through iterative corrections, ensuring accuracy while avoiding unnecessary computational complexity. The use of series expansions and convolution sums allows precise representation of nonlinear terms, making VIM a valuable tool in computational mathematics.

Overall, this study reinforces the importance of parameter sensitivity in nonlinear systems and establishes VIM as a robust methodology for tackling differential equations. These findings have significant implications for applications such as reaction-diffusion models, pattern formation, and self-organized structures in complex systems.

References

[1] R. J. Field and R. M. Noyes, *Oscillations in chemical systems. IV. Limit cycle behavior in a model of the Belousov-Zhabotinsky reaction*, The Journal of Chemical Physics, 60(5)(1974), 1877–1884.

[2] Y. Kuramoto, *Chemical Oscillations, Waves, and Turbulence*, Springer, Berlin, (1984).

[3] I. R. Epstein and J. A. Pojman, *An Introduction to Nonlinear Chemical Dynamics: Oscillations, Waves, Patterns, and Chaos*, Oxford University Press, Oxford, (1998).

[4] V. Petrov, Q. Ouyang and H. L. Swinney, *Resonant patterns in a chemical reaction*, Nature, 388(1997), 655–657.

[5] M. Bär and M. Eiswirth, *Turing structures and wave patterns in reaction-diffusion systems*, Physica D, 86(1996), 59-75.

[6] J.-H. He, *Variational iteration method – a kind of non-linear analytical technique: some examples*, International Journal of Non-Linear Mechanics, 34(4)(1999), 699–708.

[7] J.-H. He, *Variational iteration method – Some recent results and new interpretations*, Journal of Computational and Applied Mathematics, 207(1)(2007), 3–17.