

# Simulating the Dynamics of Programmed Cell Death

1<sup>st</sup> Ziad Osama

*Faculty of Engineering*

*Cairo University*

ziad.ebrahim04@eng-st.cu.edu.eg

2<sup>nd</sup> Anas Mohamed

*Faculty of Engineering*

*Cairo University*

anas.bayoumi05@eng-st.cu.edu.eg

3<sup>rd</sup> Ahmed Mahmoud

*Faculty of Engineering*

*Cairo University*

ahmed.abdelzaher04@eng-st.cu.edu.eg

4<sup>th</sup> Hassan Badawy

*Faculty of Engineering*

*Cairo University*

hassan.badawy05@eng-st.cu.edu.eg

5<sup>th</sup> Mohamed Ehab

*Faculty of Engineering*

*Cairo University*

mohamed.mohamed056@eng-st.cu.edu.eg

6<sup>th</sup> Menna Atef

*Faculty of Engineering*

*Cairo University*

menna.mahmoud06@eng-st.cu.edu.eg

7<sup>th</sup> Engy Mohamed

*Faculty of Engineering*

*Cairo University*

engy.elsarta05@eng-st.cu.edu.eg

8<sup>th</sup> Nada Mostafa

*Faculty of Engineering*

*Cairo University*

Nada.Morad05@eng-st.cu.edu.eg

9<sup>th</sup> Saga Sadek

*Faculty of Engineering*

*Cairo University*

saga.soliman05@eng-st.cu.edu.eg

**Abstract**—Accurately simulating complex biological processes like apoptosis is crucial for advancing biomedical research. In this work, we conduct a comprehensive evaluation of numerical methods for solving ordinary differential equations (ODEs), with a particular emphasis on comparing fixed-step and adaptive step-size approaches. Specifically, we develop and implement three classical solvers from scratch: the first-order Euler method, the fourth-order Runge-Kutta (RK4) method, and the adaptive Runge-Kutta-Fehlberg (RKF45) method. These algorithms are applied to a six-dimensional ODE model representing the dynamics of apoptosis. The adaptive RKF45 method, known for its high accuracy, is used as a reference to quantitatively assess the performance of the fixed-step methods. Our findings demonstrate the advantages of adaptive step-size control in terms of efficiency and reliability, while also providing a clear discussion of the trade-offs between computational cost, accuracy, and algorithmic complexity.

Furthermore, we explore a modern machine learning approach by employing Physics-Informed Neural Networks (PINNs) to solve the same ODE system, and we compare its results to those obtained from traditional numerical solvers.

**Keywords:** Apoptosis, ODE, Euler Method, Runge-Kutta, Numerical Analysis, Systems Biology, Physics-Informed Neural Networks, Machine Learning.

## I. INTRODUCTION

Apoptosis, or programmed cell death, is a crucial biological process that maintains cellular health by eliminating damaged or unnecessary cells. Disruptions in apoptosis can lead to diseases such as cancer or neurodegeneration. Mathematical models of apoptosis, typically formulated as systems of ordinary differential equations (ODE), are crucial to understanding the underlying cellular mechanisms [3]. The choice of numerical solver to integrate these equations profoundly impacts the simulation's accuracy and efficiency. Although fixed-step methods are foundational, they can be inefficient. They must

use a single small step size throughout the entire simulation, even in regions where the solution is changing slowly.

This paper explores a more advanced approach by implementing and comparing four distinct numerical solvers.

- 1) The **Runge-Kutta-Fehlberg (RKF45) method**, a modern adaptive step solver that will serve as our high accuracy baseline.
- 2) The fixed-step **fourth order Runge-Kutta (RK4) method**.
- 3) The fixed-step **explicit Euler method**.
- 4) A **Physics-Informed Neural Network (PINN)** machine learning approach.

By implementing all four from scratch, we provide a clear demonstration of their underlying logic and a quantitative comparison of their performance.

## II. LITERATURE REVIEW

The computational study of apoptosis typically relies on solving Ordinary Differential Equation (ODE) models, such as the one by Laise et al. The choice of numerical solver is critical. Foundational fixed-step methods include the simple first-order Euler method and the highly accurate fourth-order Runge-Kutta (RK4) method. A major advancement was the development of adaptive step-size solvers like the Runge-Kutta-Fehlberg (RKF45), which dynamically adjust the step size to control error, ensuring both efficiency and reliability. More recently, machine learning approaches like Physics-Informed Neural Networks (PINNs) have emerged as a powerful, mesh-free alternative for solving ODEs and tackling inverse problems in systems biology. This paper provides a direct performance benchmark of the foundational numerical

methods (Euler, RK4, and RKF45) as well as a PINN-based approach that is central to this evolving field.

### III. METHODOLOGY

#### A. ODE Model of Apoptosis

The apoptosis model consists of six coupled nonlinear ODEs that describe the concentration dynamics of hypoxia-inducible factor ( $y_{hif}$ ), oxygen ( $y_{o2}$ ), coactivator p300 ( $y_{p300}$ ), tumor suppressor p53 ( $y_{p53}$ ), caspase ( $y_{casp}$ ), and potassium ions ( $y_{kp}$ ):

$$\frac{dy_{hif}}{dt} = a_{hif} - a_3 y_{o2} y_{hif} - a_4 y_{hif} y_{p300} - a_7 y_{p53} y_{hif} \quad (1)$$

$$\frac{dy_{o2}}{dt} = a_{o2} - a_3 y_{o2} y_{hif} + a_{11} y_{hif} y_{p300} - a_{11} y_{o2} \quad (2)$$

$$\frac{dy_{p300}}{dt} = -a_4 y_{hif} y_{p300} - a_5 y_{p300} y_{p53} + a_8 \quad (3)$$

$$\frac{dy_{p53}}{dt} = a_{p53} - a_5 y_{p300} y_{p53} - a_9 y_{p53} \quad (4)$$

$$\frac{dy_{casp}}{dt} = a_9 y_{p53} + a_{12} - a_{13} y_{casp} \quad (5)$$

$$\frac{dy_{kp}}{dt} = -a_{10} y_{casp} y_{kp} + a_{11} y_{o2} - a_{14} y_{kp} \quad (6)$$

This system captures the biochemical regulation of apoptosis, showing how oxygen and genetic regulators control caspase activation and potassium ion loss—hallmarks of programmed cell death.

#### B. Parameter Settings

All simulations are run over a time interval of  $t \in [0, 100]$ . The system is solved using the initial condition  $\mathbf{y}_0 = [1, 0, 0, 0, 0, 0]$ , which represents a hypoxic state where only the HIF-1 factor is initially active. The model parameters that govern the reaction rates are adopted from the original study and are as follows (all values in appropriate units):

$$\begin{array}{llll} a_{hif} = 1.52 & a_{o2} = 1.8 & a_{p53} = 0.05 & a_3 = 0.9 \\ a_7 = 0.7 & a_8 = 0.06 & a_9 = 0.1 & a_{10} = 0.7 \\ a_4 = 0.2 & a_5 = 0.001 & a_{11} = 0.2 & a_{12} = 0.1 \\ a_{13} = 0.1 & a_{14} = 0.05 & & \end{array}$$

The numerical methods used to solve this system are detailed in the subsequent sections.

#### IV. THE BASELINE SOLVER: RUNGE-KUTTA-FEHLBERG (RKF45) METHOD

##### A. Logic and Concept

To establish a high-accuracy baseline solution, we implement the adaptive Runge-Kutta-Fehlberg (RKF45) method. RKF45 is an embedded Runge-Kutta method, meaning at each step it calculates two different approximations of the next point: a 4th-order estimate ( $\mathbf{y}^{(4)}$ ) and a more accurate 5th-order estimate ( $\mathbf{y}^{(5)}$ ). This is achieved efficiently using a shared set of six function evaluations. The difference between these two results provides a direct estimate of the local truncation error ( $ee = \|\mathbf{y}^{(5)} - \mathbf{y}^{(4)}\|$ ). This error is then

compared to a user-defined tolerance. If the error is too large, the step is rejected and re-calculated with a smaller step size,  $h$ . If the error is acceptable, the step is taken, and  $h$  may be increased for the next step. This adaptive control ensures both accuracy and efficiency.

##### B. Results

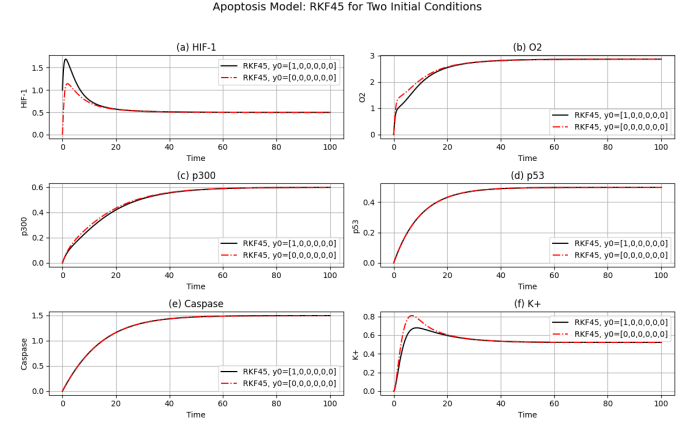


Fig. 1. Model solution using RKF45 Method.

#### V. THE 4TH-ORDER RUNGE-KUTTA (RK4) METHOD

##### A. Logic and Concept

To achieve a higher degree of accuracy, we implement the 4th-Order Runge-Kutta (RK4) method. This is an explicit, single-step solver that significantly improves upon the Euler method by sampling the derivative (the slope) at four strategic points within each time step. These points include the beginning, two distinct estimates at the midpoint, and the end of the interval.

By computing a weighted average of these four slope estimates, the RK4 method creates a much more accurate projection to the next point. This process effectively corrects for the curvature of the solution path, which is a major source of error in first-order methods. The iterative formula is given by:

$$\mathbf{y}_{n+1} = \mathbf{y}_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \quad (7)$$

where  $h$  is the step size and the terms  $k_1, k_2, k_3$ , and  $k_4$  are the four calculated slope estimates. The method's global error is proportional to  $h^4$ , providing a substantial gain in accuracy and making it a standard choice for solving non-stiff ODE problems.

##### B. Results

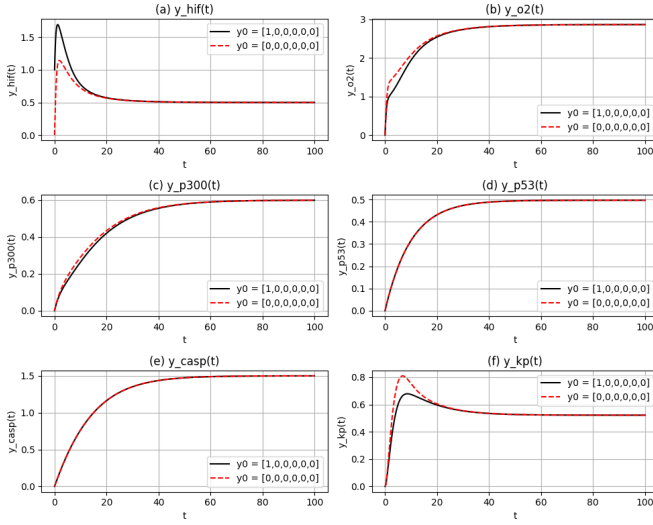


Fig. 2. Model solution using Runge-Kutta Method.

## VI. THE EXPLICIT EULER METHOD

### A. Logic and Concept

The Euler method is the simplest explicit first-order numerical scheme. It approximates the solution by taking a linear step forward using the slope at the current point. The method is defined by the iterative formula:

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h \cdot \mathbf{f}(t_n, \mathbf{y}_n) \quad (8)$$

Although easy to implement, its local truncation error is proportional to  $h^2$  and its global error is proportional to  $h$ , making it the least accurate of the methods tested. It is also only conditionally stable, requiring a small step size to avoid divergence.

### B. Results

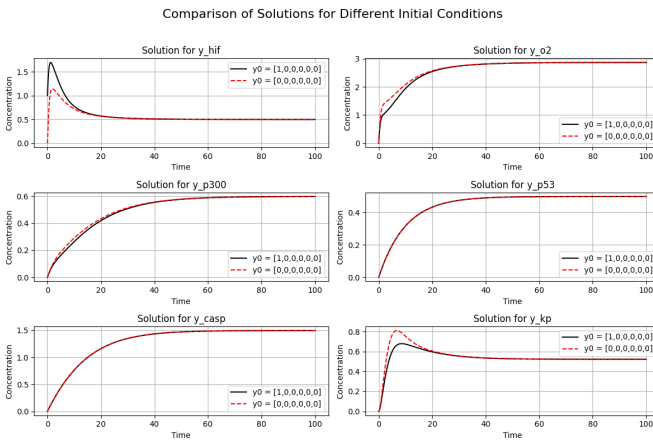


Fig. 3. Model solution using Euler Method.

## VII. PHYSICS-INFORMED NEURAL NETWORKS (PINNs) METHOD

### A. Logic and Concept

In addition to classical numerical solvers, we implemented a machine learning-based approach using Physics-Informed Neural Networks (PINNs) to solve the apoptosis ODE system. PINNs are a class of neural networks that incorporate the underlying physical laws (in this case, ODEs) directly into the loss function during training. This allows the network to learn a continuous differentiable function that approximates the solution over the entire time domain.

The PINN is trained by minimizing a composite loss function that penalizes both the deviation from the ODE system (residual loss) and the mismatch with initial conditions (initial loss). The network takes time  $t$  as input and outputs the predicted concentrations for all six variables. Automatic differentiation is used to compute the derivatives required for the ODE residuals.

This approach is mesh-free and can generalize well, even with sparse data, making it a promising tool for both forward and inverse problems in systems biology. The results are visualized in a graph that shows the difference between the PINN and RK4 solutions, highlighting how closely the PINN can approximate the classical numerical method across the time domain.

### B. Results

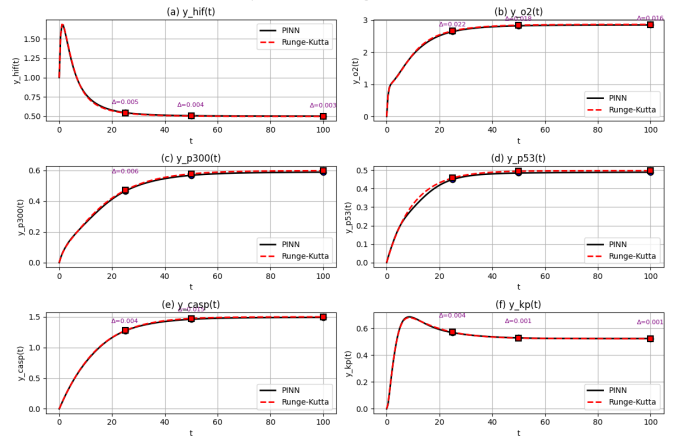


Fig. 4. Comparison between PINN and RK4 solutions for the apoptosis model.

## VIII. FUTURE WORK

This comparative study of classical numerical methods and machine learning-based PINNs provides a strong foundation for further exploration. A key area for future work is the extension of the PINN framework to solve inverse problems, such as estimating unknown biological parameters (e.g., the  $a_i$  constants) from sparse experimental data. Additionally, exploring hybrid approaches that combine the strengths of traditional solvers and neural networks could provide valuable insight into the evolving landscape of computational biology.

## IX. SIMULATION RESULTS AND DISCUSSION

Table I reveals that while RK4 maintains relatively low errors across all variables (peaking at 1.956% for  $y_{p300}$ ), the Explicit Euler method shows remarkably small errors (on the order of  $10^{-6}\%$ ), and PINN exhibits higher but still reasonable errors (peaking at 4.34% for  $y_{p53}$ ).

TABLE I  
PERCENTAGE ERRORS FOR EACH VARIABLE ACROSS METHODS

Variable	RK4	Explicit Euler	PINN
$y_{hif}$	0.512%	2.7234e-07%	2.54%
$y_{o2}$	0.499%	1.9473e-06%	2.12%
$y_{p300}$	1.956%	1.4624e-06%	3.49%
$y_{p53}$	0.013%	8.4446e-08%	4.34%
$y_{casp}$	0.001%	1.1656e-06%	0.76%
$y_{kp}$	0.599%	1.7777e-07%	1.61%

Table II highlights the trade-offs between computational efficiency and accuracy. RKF45 emerges as the most efficient method with both rapid execution (0.042 sec) and good accuracy (max error 1.421%). The PINN approach, while offering the advantage of being mesh-free, requires significantly more computational time (20.81 sec) and has higher errors compared to traditional numerical methods.

TABLE II  
OVERALL PERFORMANCE COMPARISON OF METHODS

Method	Execution Time	Max Error	Min Error
RKF45	0.042 sec	1.421%	0.271%
RK4	0.939 sec	1.956%	0.001%
Explicit Euler	0.189 sec	1.940%	0.015%
PINN	20.81 sec	4.34%	0.76%

The quantitative results in Table III show that all four methods—RKF45, RK4, Euler, and the machine learning-based PINN (ML)—produce very similar solution values for each variable at the selected time points ( $t = 25, 50, 75, 100$ ). Tables I and II provide additional insights into the performance characteristics of each method.

TABLE III  
COMPARISON OF SOLUTION VALUES AT DIFFERENT TIME POINTS (INITIAL CONDITION: [1,0,0,0,0,0])

Method	Time	$y_{hif}$	$y_{o2}$	$y_{p300}$	$y_{p53}$	$y_{casp}$	$y_{kp}$
RKF45	25	0.5407	2.6699	0.4732	0.4587	1.2789	0.5693
	50	0.5037	2.8410	0.5773	0.4941	1.4711	0.5266
	75	0.4996	2.8618	0.5950	0.4968	1.4943	0.5223
	100	0.4991	2.8646	0.5978	0.4970	1.4968	0.5219
RK4	25	0.5420	2.6645	0.4706	0.4576	1.2735	0.5706
	50	0.5038	2.8407	0.5771	0.4940	1.4708	0.5267
	75	0.4996	2.8618	0.5950	0.4968	1.4943	0.5223
	100	0.4991	2.8646	0.5978	0.4970	1.4968	0.5219
Euler	25	0.5419	2.6647	0.4707	0.4576	1.2737	0.5705
	50	0.5038	2.8408	0.5771	0.4940	1.4709	0.5267
	75	0.4996	2.8618	0.5950	0.4968	1.4943	0.5223
	100	0.4991	2.8646	0.5978	0.4970	1.4968	0.5219
ML	25	0.5469	2.6449	0.4661	0.4501	1.2718	0.5664
	50	0.5081	2.8231	0.5672	0.4841	1.4567	0.5278
	75	0.4961	2.8538	0.5916	0.4933	1.4908	0.5188
	100	0.5027	2.8491	0.4887	0.4887	1.4878	0.5228

## X. CONCLUSION

We implemented and compared the Explicit Euler, 4th-Order Runge-Kutta, adaptive Runge-Kutta-Fehlberg (RKF45), and Physics-Informed Neural Network (PINN) methods for solving an ODE model of apoptosis. Using our RKF45 implementation as a high-precision baseline, we found that all four methods produced stable and consistent results. Both the 4th-Order Runge-Kutta and PINN offered a good balance of accuracy and efficiency. Overall, our results show that even simple numerical methods and modern machine learning approaches can reliably solve well-behaved biological systems.

## REFERENCES

- [1] D. R. Green and G. Evan, "A matter of life and death," *Cancer Cell*, vol. 1, no. 1, pp. 19-30, 2002.
- [2] E. T. Bagci, Y. Vodovotz, T. R. Billiar, et al., "Bistability in apoptosis," *Biophysical Journal*, vol. 90, no. 5, pp. 1546-1559, 2006.
- [3] P. Laise, D. Fanelli, and A. Arcangeli, "A dynamical model of apoptosis and its role in tumor progression," *Communications in Nonlinear Science and Numerical Simulation*, vol. 17, no. 4, pp. 1795-1804, 2012.
- [4] J. C. Butcher, *Numerical Methods for Ordinary Differential Equations*. John Wiley & Sons, 2016.
- [5] L. F. Shampine and M. W. Reichelt, "The MATLAB ODE suite," *SIAM Journal on Scientific Computing*, vol. 18, no. 1, pp. 1-22, 1997.