ISSN: 3023 -8951

CHAOS AND FRACTALS

VOLUME 1, ISSUE 1, JULY 2024

AN INTERDISCIPLINARY JOURNAL OF NONLINEAR SCIENCE

ADBA

https://journals.adbascientific.com/chf

ISSN: 3023-8951

Chaos and Fractals

Volume: 1 – Issue No: 1 (July 2024)

EDITORIAL BOARD

https://journals.adbascientific.com/chf/about/editorialTeam

ISSN: 3023-8951

Chaos and Fractals
Volume: 1 – Issue No: 1 (July 2024)

CONTENTS

1 **Dianorré Tokoue Ngatcha, Buris Peggy Ndemanou, Alain Francis Talla, Serge Gervais Ngueuteu Mbouna, Sifeu Takougang Kingni**
Numerical Exploration of a Network of Nonlocally Coupled Josephson Junction Spurred by Wien Bridge Oscillators **(Research Article)**

6 **Harun Emre Kıran**
A Novel Chaos-Based Encryption Technique with Parallel Processing Using CUDA for Mobile Powerful GPU Control Center **(Research Article)**

19 **Cem Özkurt**
Comparative Analysis of State-of-the-Art Q&A Models: BERT, RoBERTa, DistilBERT, and ALBERT on SQuAD v2 Dataset **(Research Article)**

31 **Jules Metsebo , Boubakary Abdou, Dianorré Tokoue Ngatcha, Isidore Komofor Ngongiah, Paul Didier Kamdem Kuate, Justin Roger Mboupda Pone**
Analysis of a Resistive-Capacitive Shunted Josephson Junction with Topologically Nontrivial Barrier Coupled to a RLC Resonator (**Research Article**)

38 **Berkay Emin, Mustafa Yaz**
Digital Implementation of Chaotic Systems Using Nvidia Jetson AGX Orin and Custom DAC Converter
(**Research Article**)

https://journals.adbascientific.com/chf/issue/view/4

ADBA Chaos and Fractals
Cutting-Edge Scientific Solutions

e-ISSN: 3023-8951
RESEARCH ARTICLE
Vol.1 / No.1 / 2024 / pp.1-5
https://doi.org/10.69882/adba.chf.2024071

Numerical Exploration of a Network of Nonlocally Coupled Josephson Junction Spurred by Wien Bridge Oscillators

Dianorré Tokoue Ngatcha[*,1], **Buris Peggy Ndemanou**[α,2], **Alain Francis Talla**[α,3], **Serge Gervais Ngueuteu Mbouna**[β,4] and **Sifeu Takougang Kingni**[α,5]

[*]University of Douala, National Higher Polytechnic School of Douala, Department of Automotive and Mechatronics Engineering, Box 24, 2701 Douala, Cameroon, [α]University of Maroua, Faculty of Mines and Petroleum Industries, Department of Mechanical, Petroleum and Gas Engineering, P.O. Box 46, Maroua, Cameroon, [β]University of Yaoundé I, Faculty of Science, Laboratory of Modelling and Simulation in Engineering, Biomimetics and Prototypes, P.O. 812, Yaoundé, Cameroon.

ABSTRACT This paper is devoted to the numerical investigation of the collective dynamics of nonlocally coupled Josephson junctions (JJ) spurred by Wien bridge oscillators (JJSWBOs). A single JJSWBO displays monostable and bistable chaotic characteristics, bistable period-3-oscillations and coexistence between regular and chaotic characteristics. In order to investigate the collective dynamics of the considered network, the local dynamics is set in the bistable regime which includes periodic and chaotic dynamics. The initial conditions for the network dynamics are specially prepared such that the networks units are randomly distributed on these two types of dynamical behaviors. As the value of the coupling strength increases, the dynamics of the network of JJSWBOs undergoes a transition from complete incoherence to coherent travelling waves via a chimera state.

KEYWORDS
Josephson junction
Wien bridge oscillator
Chaotic and coexisting characteristics
Nonlocal coupling
Coherent and incoherent behaviors
Chimera state

INTRODUCTION

Many phenomena in nature result from the intricate interaction between the dynamical characteristics of tremendous elements. For example, the cognitive function of the brain and certain brain disorders such as epilepsy and Parkinson's disease are the macroscopic manifestation of the interaction between the neurons. The most investigated collective dynamical behavior of interacting elements is synchronization, which can be simply defined as the entrainment of rhythms of these interacting elements (Pikovsky *et al.* 2001). The study of synchronization is of paramount importance since it is ubiquitous in the functioning of natural and manufactured systems.

Among the plethora of synchronization phenomena, the chimera state has attracted a lot of attention since its discovery in the early 2000s (Kuramoto and Battogtokh 2002). The chimera state is an intriguing partial synchronization state arising in networks of symmetrically coupled identical systems and characterized by coexisting domains of coherent and incoherent systems. The chimera state was first discovered and mathematically studied in networks of nonlocally coupled phase oscillators (Kuramoto and Battogtokh 2002; Abrams and Strogatz 2004). Since then, a plethora of related patterns have been unveiled in diverse networks of diverse systems, including limit-cycle oscillators, chaotic oscillators, excitable systems, and multistable systems (for a review, see (Zakharova 2020; Parastesh *et al.* 2021; Schöll 2016)). In addition, chimera states were uncovered in experimental setups (Hagerstrom *et al.* 2012; Tinsley *et al.* 2012).

The study of collective dynamics of coupled systems of Josephson junctions is a subject that has been tackled in recent years. Many works on synchronization in arrays of Josephson junctions were carried out (Ngongiah *et al.* 2023; Ramakrishnan *et al.* 2021; Filatrella *et al.* 1992). Very recently, a few works reported the observation of chimeralike states in networks of Josephson junctions (Mishra *et al.* 2017a,b). Thus, the investigation of chimera states in coupled systems of Josephson junctions remains an open problem.

Manuscript received: 20 July 2024,
Revised: 27 July 2024,
Accepted: 30 July 2024.

[1]tokouengatcha1@gmail.com (**Corresponding author**)
[2]ndemanoupeggy@gmail.com
[3]alainfrancis.aft@gmail.com
[4]gsmngueut@gmail.com
[5]stkingni@gmail.com

Besides, bi- or multi-stability has proved to be a key ingredient for the formation of chimera states (Yeldesbay *et al.* 2014). Diverse chimera and chimeralike states have been observed in diverse networks of multistable systems (Dudkowski *et al.* 2016; Mishra *et al.* 2015; Shepelev *et al.* 2017; Lei *et al.* 2022; Ramamoorthy *et al.* 2022; Zhang *et al.* 2024; Nganso *et al.* 2023). In the present study, motivated by the fact that for some parameter values, the JJSWBO is multistable, the authors of this paper intend to investigate chimera states in a network of many JJSWBOs.

DYNAMICAL CHARACTERISTICS OF JJSWBO AND NETWORK OF NONLOCALLY COUPLED JJSWBOS

The JJSWBO designed by coupling through, a gain a resistive-capacitive shunted JJ is described by the given set of equations (Sriram *et al.* 2023):

$$\frac{dV_1}{dt} = -\frac{k}{2}\left[(|V_1+k|-|V_1-k|)-V_2-\beta_R V_1-\alpha\sin(\phi)\right], \quad (1a)$$

$$\frac{dV_2}{dt} = \beta_c\left[\frac{k}{2}(|V_1+k|-|V_1-k|)-V_2\right], \quad (1b)$$

$$\frac{d\phi}{dt} = \gamma V_1. \quad (1c)$$

where v_1 and v_2 are the voltages, ϕ the phase difference of the JJ, and k, α, β_R, β_c, and γ are the system parameters. The dynamical characteristics of JJSWBO are illustrated by the phase plane projections of Fig. 1.

Monostable, stable, and chaotic characteristics of different presentations are shown in Fig. 1(a-e), bistable period-3 oscillations for specific control parameters and varying recipient states as depicted in Fig. 1(f), bistable chaotic attractors for chosen parameters and different initial states as elaborated in Fig. 1(g), coexistence between regular and chaotic characteristics for some fixed values of control parameters and different starting conditions as captured in Fig. 1(h1 and h2). The coexistence of periodic and chaotic characteristics shown in Fig. 1(h1 and h2) is further illustrated by the basin of attraction in Fig. 2. The attraction basin of Fig. 2 shows the regions of chaotic characteristics painted in magenta and the regions of periodic characteristics painted in green.

Searching for the chimera states, a lot of coupling functions is tested and it is found that chimera states emerge in networks of nonlocally coupled JJSWBOs with direct and cross terms in the coupling function which is applied on the v_1-variable differential equation. A large network of nonlocally coupled JJSWBOs is considered, where each network unit is coupled to its $2P$ nearest neighbors, with P neighbors at either direction around the considered unit. Thus, the dynamics of a given node j of the considered network is described by the following set of differential equations

$$\frac{dv_j^1}{dt} = -\frac{k}{2}\left(\left|v_j^1+k\right|-\left|v_j^1-k\right|\right)-v_j^2-\beta_R v_j^1 - \alpha\sin(\phi^j)+\sigma\sum_{i=j-P}^{j+P}\left[a(v_i^1-v_j^1)+b(\phi^i-\phi^j)\right] \quad (2a)$$

$$\frac{dv_j^2}{dt} = \beta_c\left[\frac{k}{2}\left(\left|v_j^1+k\right|-\left|v_j^1-k\right|\right)-v_j^2\right], \quad (2b)$$

$$\frac{d\phi_j}{dt} = \gamma v_j^1, \quad (2c)$$

where $j = 1,2,\ldots,N$, N is the size of the network, $\sigma_1 = a\sigma$ and $\sigma_2 = b\sigma$ are real numbers that denote the coupling strengths. In addition, periodic boundary conditions are assumed. For the formation of chimera states in the network, α must be positive and β negative. The combination of direct and cross couplings proved to induce the occurrence of chimeralike states and other symmetry-breaking states, namely, oscillation death states in a network of globally coupled JJs (Mishra *et al.* 2017b). It was shown in (Nganso *et al.* 2023) that such a competitive coupling is also responsible for the formation of chimera and solitary states in networks of van der Pol oscillators.

In order to obtain chimera states, initial conditions very often play a pivotal role Zakharova (2020); Parastesh *et al.* (2021); Schöll (2016). The historically first observed chimera state was obtained with specially prepared initial conditions (Kuramoto and Battogtokh 2002). In the present paper, if the dynamics of all network nodes are launched from the basin of attraction of the periodic attractor (green strips in Fig. 2), they converge to the same dynamical state, and even a phase lag between the periodic oscillations of the different nodes is not observed. Thus, in the perspective of investigating symmetry-breaking states, we managed to get initial conditions disseminated randomly in the basins of attraction of the periodic and chaotic attractors. To do so, we considered $\phi_j(0) = 8$ for all nodes (in accordance with the result displayed in Fig. 2), $v_j^1(0)$ and $v_j^2(0)$ are randomly distributed in $[-0.25, 0.25]$ and $[-6, -5]$, respectively.

The set of equations describing the considered network [i.e., Eq. (2)] is solved numerically with the above initial conditions, the parameters used in Fig. 2 for the local dynamics, and $a = 0.08$, $b = -0.8$, and $P = 20$ for the coupling term. The coupling parameter σ is considered as a control parameter for the investigation of the collective dynamics of the coupled oscillators. The results of the numerical simulation of Eq. (2) are shown in Fig. 3 for an increasing value of the coupling parameter σ. Apart from spatiotemporal plots of the different collective dynamical states obtained when varying σ, the corresponding mean phase velocity profiles $\{\omega_j\}, j = 1,2,\ldots,N$ are also displayed. The mean phase velocity ω_j of a given network node j is evaluated numerically from the time series of v_j^1 as follows:

$$\omega_j = \frac{2\pi M_j}{\Delta t}, \quad (3)$$

where M_j is the number of oscillations performed by v_j^1 during the time interval Δt Omelchenko *et al.* (2013).

The mean phase velocity is a quantitative measure widely used for the characterization of certain chimera states and solitary states. Indeed, in case of phase chimera, the mean phase velocity profile presents flat pieces which correspond to coherent parts of the chimera pattern and arc-shaped pieces which correspond to incoherent parts (Kuramoto and Battogtokh 2002; Abrams and Strogatz 2004). For a solitary state, the mean phase velocity profile is overall flat and it involves some singularities which break away from the flat background—the flat background corresponds to the common frequency of synchronous nodes and the singularities correspond to solitary nodes (Rybalova *et al.* 2019).

Fig. 3(a) confirms that the isolated nodes ($\sigma = 0$, i.e., the network nodes are uncoupled) are randomly distributed on two types of dynamical behaviors: the periodic and chaotic dynamics. All the oscillators exhibiting the periodic behavior have the same frequency and the oscillators exhibiting chaotic dynamics have disparate frequencies [see $\{\omega_j, j = 1,2,\ldots,N\}$ in Fig. 3(a)]. As the value of the coupling parameter σ increases, the dynamics of the network switches from complete incoherence [Fig. 3(b)] to coherent

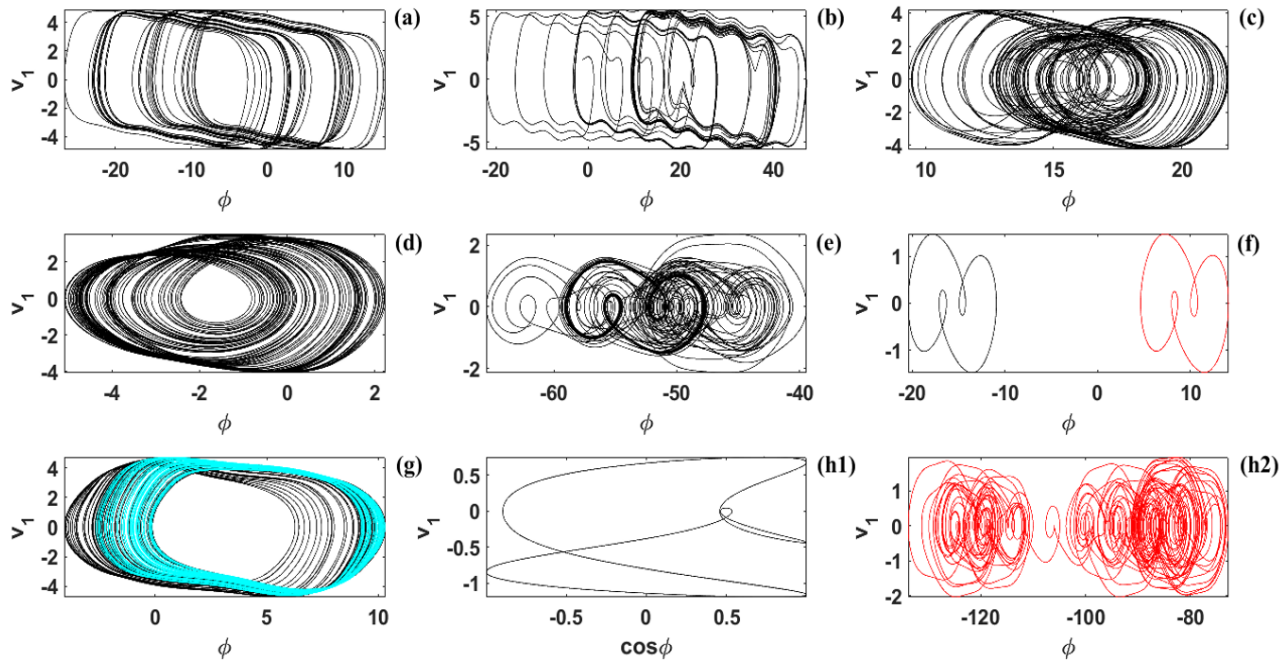


Figure 1 Evolution of JJSWBO described by system (1) in the plane (v_1, ϕ) for varying α, β_R, and γ: **(a)** $\alpha = 0.2$, $\beta_R = 0.1$, $\gamma = 0.9$, **(b)** $\alpha = 0.9$, $\beta_R = 0.1$, $\gamma = 0.9$, **(c)** $\alpha = 0.45$, $\beta_R = 0.468$, $\gamma = 0.9$, **(d)** $\alpha = 0.45$, $\beta_R = 0.535$, $\gamma = 0.9$, **(e)** $\alpha = 0.45$, $\beta_R = 0.8$, $\gamma = 4.05$, **(f)** $\alpha = 0.45$, $\beta_R = 0.258$, $\gamma = 0.9$, and **(h1, h2)** $\alpha = 0.45$, $\beta_R = 0.8$, $\gamma = 5.037$. The remaining system parameters are limited to $k = 3.2$ and $\beta_c = 2.5$. The recipient conditions include: $(v_1(0), v_2(0), \phi(0)) = (0, 0, 1)$ for black lines, $(v_1(0), v_2(0), \phi(0)) = (0, 8, 0)$ for red lines, and $(v_1(0), v_2(0), \phi(0)) = (1, 1, 1)$ for cyan lines.

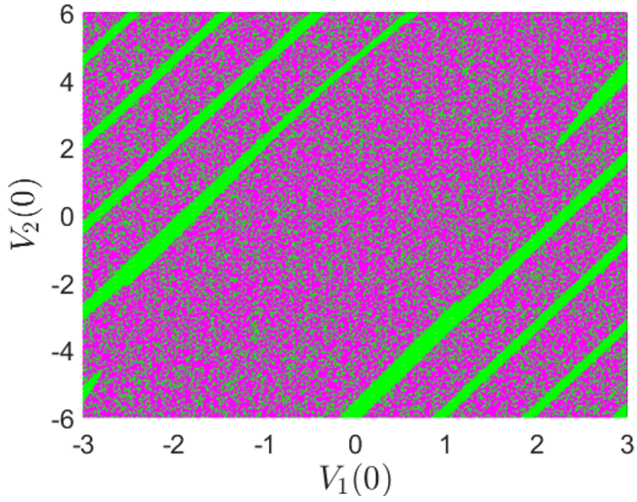


Figure 2 Overview of the basin of attraction of system (1) in the coordinate space for $(-3 \leq V_1(0) \leq 3, -6 \leq V_2(0) \leq 6)$ with parameter values $\phi(0) = 8$, $\alpha = 0.45$, $\beta_R = 0.8$, $\gamma = 5.037$, $k = 3.2$ and $\beta_c = 2.5$.

traveling waves [Fig. 3(d)] via a chimera state [Fig. 3(c)]. The mean phase velocity profile in Fig. 3(b) is completely erratic (complete incoherence), while it is flat in Fig. 3(d) as a proof of complete coherence. Fig. 3(c) shows a peculiar form of coherence-incoherence pattern where we can clearly distinguish between small incoherent parts and coherent parts in which a larger number of nodes are self-organized in the form of traveling waves. The phase chimera nature of this coherence-incoherence state is demonstrated by the mean phase velocity profile in Fig. 3(c), where the nodes in the coherent clusters are characterized by the same mean phase velocity, and the incoherent nodes have disparate phase velocities that break away from the common phase velocity of the coherent nodes. Furthermore, the network nodes belonging to the incoherent parts of the chimera state exemplifies in Fig. 3(c) exhibit irregular oscillations while the nodes of the coherent parts evolve on periodic oscillations [see Fig. 4(a)]. Fig. 4(b) shows that the oscillations of the nodes of the traveling waves pattern illustrated in Fig. 3(d) are periodic.

For an overall view of the dynamical behavior of the considered network of JJSWBOs for the parameter values mentioned above and the control parameter σ, we resort to the strength of incoherence, a global order parameter that helps to characterize diverse collective states in the coupled system, depending on their degree of incoherence. The strength of incoherence is given as follows Gopal *et al.* (2018):

$$S = 1 - \frac{1}{M} \sum_{m=1}^{M} H(\delta - s_m), \tag{4}$$

where $H(\cdot)$ stands for the Heaviside function, δ is a small threshold, and

$$s_m = \left\langle \sqrt{\frac{1}{n} \sum_{i=(m-1)n+1}^{mn} [Z_i - \langle Z \rangle_m]^2} \right\rangle_t \tag{5}$$

where $n = N/M$, M is the number of bins of nodes of equal size n, $m = 1, 2, \ldots, M$,

$$z_i = v_i^1 - v_{i+1}^1, \quad \langle z \rangle_m = \frac{1}{n} \sum_{i=(m-1)n+1}^{mn} z_i, \quad \text{and} \quad \langle \cdot \rangle_t$$

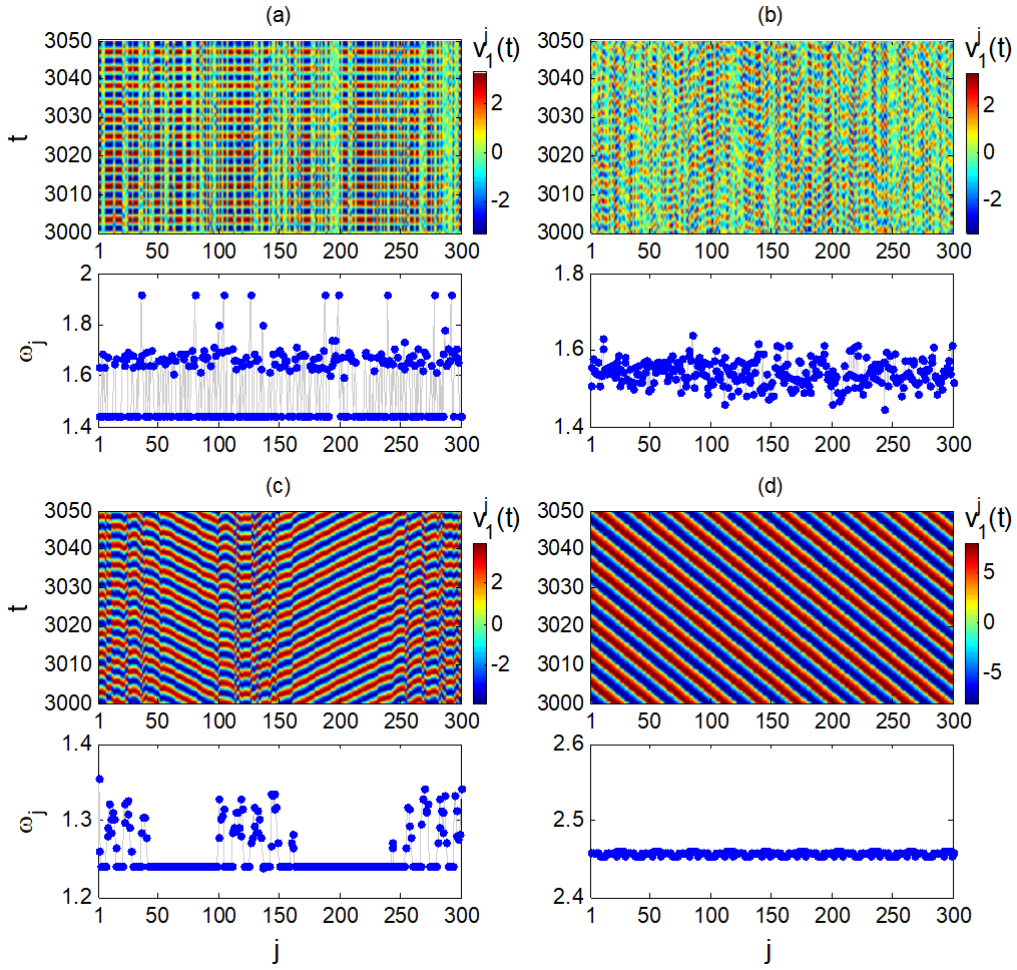


Figure 3 Prominent collective states exhibited by the network of nonlocally coupled JJSWBOs [Eq. (2)] for $N = 300$, $a = 0.08$, $b = -0.8$, $P = 20$, and different values of σ **(a)** $\sigma = 0.00$; **(b)** $\sigma = 0.06$; **(c)** $\sigma = 0.20$; **(d)** $\sigma = 0.60$. Other parameters are given in Fig. 2. In each subfigure, the top panel shows the spatiotemporal plot of the variable $v_1(t)$ for the observed pattern and the bottom panel shows the corresponding phase velocity profile $\{\omega_j, j = 1, 2, \ldots, N\}$.

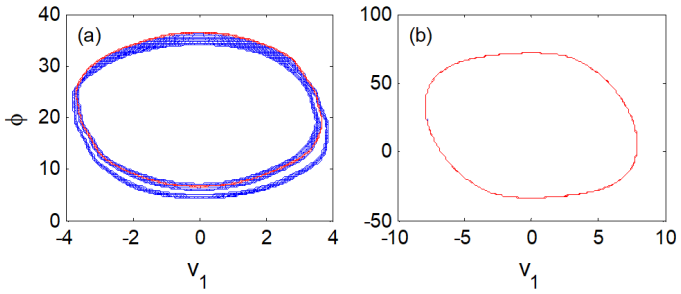


Figure 4 Phase portraits of two selected nodes ($j = 130$ and $j = 200$) of the chimera pattern and traveling waves pattern shown in Fig. 3: (a) parameters of Fig. 3(c); and (b) parameters of Fig. 3(d). The phase portrait of node 130 (respectively, 200) is displayed in blue (respectively red) line. Note that in (a) node 130 belongs to an incoherent part of the chimera state and node 200 to a coherent part. In (b) the two phase portraits are merged with one another.

stands for the time average. The value of δ is chosen such that, if the bin m is coherent, then $s_m < \delta$. Overall, $S \to 1$ for completely incoherent states, $S \to 0$ for complete coherent states, and $0 < S < 1$ for coherence-incoherence states, including chimera states. Fig. 5 shows the variation of the strength of incoherence S in the case of the network under study in this paper, when the value of the coupling parameter σ is varying. Fig. 5 confirms the transition from complete incoherence ($S \to 1$) to complete coherence ($S = 0$) via chimera states ($0 < S < 1$) when the value of σ is increasing.

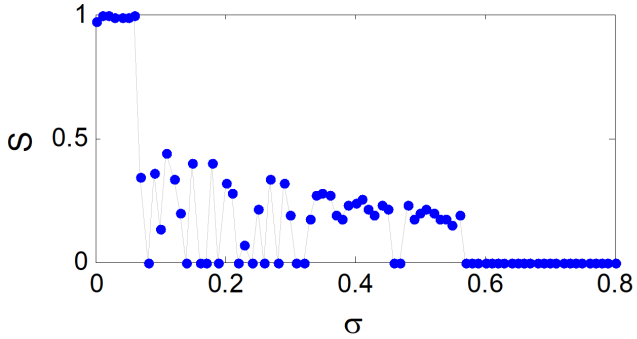


Figure 5 Strength of incoherence S versus coupling parameter σ. The parameters are given in Figs. 2 and 3.

CONCLUSION

This paper dealt with the dynamical exploration of a network of nonlocally coupled Josephson junctions spurred by Wien bridge oscillators. First, the dynamics of an isolated network unit (i.e., a Josephson junction spurred by a Wien bridge oscillator) was investigated numerically, which revealed that the local dynamics is rich, including multistability. For the investigation of the collective dynamics of the considered network, the local dynamics was set to be bistable, involving periodic and chaotic behaviors. The initial conditions were specially prepared such that the network nodes were randomly distributed on these two types of dynamical behaviors. Varying the coupling strength, the nonlocally coupled Josephson junctions spurred by Wien bridge oscillators was found to exhibit complete incoherence, coherent travelling waves and chimera behaviors.

Availability of data and material

Not applicable.

Conflicts of interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

LITERATURE CITED

Abrams, D. M. and S. H. Strogatz, 2004 Chimera states for coupled oscillators. Physical Review Letters **93**: 174102.

Dudkowski, D., Y. Maistrenko, and T. Kapitaniak, 2016 Occurrence and stability of chimera states in coupled externally excited oscillators. Chaos **26**: 116306.

Filatrella, G., G. Rotoli, N. Grønbech-Jensen, R. D. Parmentier, and N. F. Pedersen, 1992 Model studies of long josephson junction arrays coupled to a high-q resonator. Journal of Applied Physics **72**: 3179–3185.

Gopal, R., V. K. Chandrasekar, D. V. Senthilkumar, A. Venkatesan, and M. Lakshmanan, 2018 Chimera at the phase-flip transition of an ensemble of identical nonlinear oscillators. Communications in Nonlinear Science and Numerical Simulation **59**: 30–46.

Hagerstrom, A. M., T. E. Murphy, R. Roy, P. Hövel, I. Omelchenko, *et al.*, 2012 Experimental observation of chimeras in coupled-map lattices. Nature Physics **8**: 658–661.

Kuramoto, Y. and D. Battogtokh, 2002 Coexistence of coherence and incoherence in nonlocally coupled phase oscillators. Nonlinear Phenomena in Complex Systems **5**: 380–385.

Lei, Z., S. Pu, H. Zhang, C. Yao, Y. Qian, *et al.*, 2022 Bistability-induced chimeras in one-dimensional paced excitable ring with nonlocal couplings. Europhysics Letters **139**: 62001.

Mishra, A., C. Hens, M. Bose, P. K. Roy, and S. K. Dana, 2015 Chimeralike states in a network of oscillators under attractive and repulsive global coupling. Physical Review E **92**: 062920.

Mishra, A., S. Saha, P. K. Roy, M. Bose, P. Louodop, *et al.*, 2017a Coherent libration to coherent rotational dynamics via chimera-like states and clustering in a josephson junction array. Physical Review E **95**: 010201(R).

Mishra, A., S. Saha, P. K. Roy, T. Kapitaniak, and S. K. Dana, 2017b Multicluster oscillation death and chimeralike states in globally coupled josephson junctions. Chaos **27**: 023110.

Nganso, E. N., S. G. N. Mbouna, R. Yamapi, G. Filatrella, and J. Kurths, 2023 Two-attractor chimera and solitary states in a network of nonlocally coupled birhythmic van der pol oscillators. Chaos, Solitons and Fractals **169**: 113235.

Ngongiah, I., B. Ramakrishnan, G. Kuiate, A. Akgül, and B. Nana, 2023 Mechanical arm(s) driven by josephson junction circuit(s), mimicking the movement pattern of myriapods. Physica Scripta **98**: 045221.

Omelchenko, I., O. E. Omel'chenko, P. Hövel, and E. Schöll, 2013 When nonlocal coupling between oscillators becomes stronger: patched synchrony or multichimera states. Physical Review Letters **110**: 224101.

Parastesh, F., S. Jafari, H. Azarnoush, Z. Shahriari, Z. Wang, *et al.*, 2021 Chimeras. Physics Reports **898**: 1–114.

Pikovsky, A., M. Rosenblum, and J. Kurths, 2001 *Synchronization: A Universal Concept in Nonlinear Science*. Cambridge University Press, Cambridge, UK.

Ramakrishnan, B., L. Tabejieu, I. Ngongiah, S. Kingni, R. Siewe, *et al.*, 2021 Suppressing chaos in josephson junction model with coexisting attractors and investigating its collective behavior in a network. Journal of Superconductivity and Novel Magnetism **34**: 2761–2769.

Ramamoorthy, R., Z. Shahriari, H. Natiq, K. Rajagopal, and C. Li, 2022 Chimera states in coupled memristive chaotic systems: effects of control parameters. Europhysics Letters **139**: 41001.

Rybalova, E., V. S. Anishchenko, G. I. Strelkova, and A. Zakharova, 2019 Solitary states and solitary state chimera in neural networks. Chaos **29**: 071106.

Schöll, E., 2016 Synchronization patterns and chimera states in complex networks: interplay of topology and dynamics. The European Physical Journal Special Topics **225**: 891–919.

Shepelev, I. A., T. E. Vadivasova, A. V. Bukh, G. I. Strelkova, and V. S. Anishchenko, 2017 New type of chimera structures in a ring of bistable fitzhugh–nagumo oscillators with nonlocal interaction. Physics Letters A **381**: 1398–1404.

Sriram, B., D. Godonou, C. Ainamon, I. Ngongiah, and K. Rajagopal, 2023 Analysis and microncontroller implementation of josephson junction driven by wien bridge oscillator. Physica Scripta **98**: 045213.

Tinsley, M. R., S. Nkomo, and K. Showalter, 2012 Chimera and phase-cluster states in populations of coupled chemical oscillators. Nature Physics **8**: 662–665.

Yeldesbay, A., A. Pikovsky, and M. Rosenblum, 2014 Chimeralike states in an ensemble of globally coupled oscillators. Physical Review Letters **112**: 144103.

Zakharova, A., 2020 *Chimera Patterns in Networks: Interplay between Dynamics, Structure, Noise, and Delay*. Springer Nature Switzerland AG, Cham.

Zhang, H., Z. Chen, F. Liu, Z. Lei, Z. Zheng, *et al.*, 2024 Alternate attractor chimeralike states on rings of chaotic lorenz-type oscillators. New Journal of Physics **26**: 023016.

How to cite this article: Ngatcha, D. T., Ndemanou, B. P., Talla, A. F., Mbouna, S. G. N., and Kingni, S. T. Numerical Exploration of a Network of Nonlocally Coupled Josephson Junction Spurred by Wien Bridge Oscillators. *Chaos and Fractals*, 1(1), 1-5, 2024.

ADBA Chaos and Fractals
Cutting-Edge Scientific Solutions

e-ISSN: 3023-8951
RESEARCH ARTICLE
Vol.1 / No.1 / 2024 / pp.6-18
https://doi.org/10.69882/adba.chf.2024072

A Novel Chaos-Based Encryption Technique with Parallel Processing Using CUDA for Mobile Powerful GPU Control Center

Harun Emre Kıran *,[1]
*Department of Computer Engineering, Hitit University, 19030, Corum, Turkiye.

ABSTRACT Chaotic systems possess unique properties that can be leveraged for cybersecurity. These properties stem from the complex and unpredictable nature of images, which makes it challenging for systems to interpret them. When combined with CUDA, chaotic systems benefit from high-efficiency parallel processing capabilities, allowing for the rapid and secure handling of large data sets. Therefore, chaotic systems can be effectively used to securely store and conceal images. In this study, a CUDA-supported chaos-based parallel processing encryption mechanism for mobile control centers is developed. Encryption processes leverage the powerful GPU of the control center. This allows for the fast encryption and decryption of image data received from multiple devices connected to the control center via wired or wireless connections. For encryption, the Logistic Map is used to generate random numbers. Using this map, image data is subjected to XOR operations, encrypting the R, G, B, and Gray Scale channels of the images. Initially, an analysis of the numbers generated from this map is conducted, followed by a detailed explanation of the encryption technique. The technique is then applied to image data, and image analyses are performed. Finally, the performance of the encryption technique is compared with other studies, and encryption speeds are examined. The results show that the new encryption technique provides significantly fast encryption and security levels comparable to other studies. The key discovery of this research is that the devised mechanism is well-suited for parallel processing, allowing for rapid image encryption using the proposed method. For encrypting large IoT files, random number generation is initially performed, followed by statistical tests. Subsequently, encryption is executed using the developed algorithm, and security analyses are conducted. The performance of the proposed mechanism is compared with other studies in the literature, and the results from image analysis and encryption performance demonstrate that the developed mechanism can be effectively used with high security for IoT applications.

KEYWORDS
Chaos-based encryption
CUDA programming
Logistic map
Image encryption
Internet of things

INTRODUCTION

Chaos theory is a field of study that explains nonlinear phenomena that appear disordered but have an underlying order. This theory addresses two concepts: determinism and randomness. Specifically, deterministic chaos theory explains that complex systems governed by deterministic rules exhibit behaviors that appear random due to their sensitivity to initial conditions. Due to the deterministic nature of this chaos, it is used in various fields such as encryption (Clemente-Lopez *et al.* 2024), motor control (Mai *et al.* 2015), image processing (Boyraz *et al.* 2022), random number generation (Tutueva *et al.* 2020), and the Internet of Things (Kumari and Mondal 2023).

Chaos theory plays a significant role in cryptography by explaining the mathematical order behind seemingly random events, thus ensuring data security (Naik and Singh 2024). This characteristic has enabled the development of innovative encryption methods in various fields. Furthermore, the inherent randomness of chaotic systems has proven to be a valuable resource for generating strong random numbers, which are a fundamental component of many cryptographic algorithms (Man *et al.* 2021).

The arrival of the Fourth Industrial Revolution has brought the Internet of Robotic Things to the forefront by combining the capabilities of the Internet of Things with autonomous robots, initiating a new era of interconnectedness (Romeo *et al.* 2020). This transformative technology has revolutionized industries such

Manuscript received: 3 May 2024,
Revised: 28 July 2024,
Accepted: 30 July 2024.

[1]harunemrekiran@hitit.edu.tr (**Corresponding author**)

as manufacturing (Singh *et al.* 2021), agriculture (Kashyap *et al.* 2021), healthcare (Rajendran and Doraipandian 2021), education (Francisti *et al.* 2020), and surveillance (Ghosh *et al.* 2021), leading to limitations in processing power and storage capacity commonly encountered in robots that necessitate efficient methods to encrypt large data sets. To overcome this challenge, a novel discrete-time chaotic encryption mechanism is proposed to securely store and conceal large data sets in robots (Kiran *et al.* 2023; Erkan *et al.* 2023).

Compute Unified Device Architecture (CUDA), the parallel computing platform and application programming interface developed by NVIDIA, has emerged as a game changer in high-performance computing. Its ability to leverage the general-purpose processing power of Graphics Processing Units has led to significant advancements in various fields, including cryptography (Jadhav *et al.* 2023). CUDA's parallel processing capabilities have been particularly useful for accelerating computationally intensive encryption and decryption tasks, making it an attractive option for real-time encryption and decryption applications.

Today, there are many studies related to parallel encryption using CUDA, especially in conjunction with chaos. The paper presented by Bezerra *et al.* (2024) introduces a novel single-core parallel image encryption scheme based on chaotic maps, and this method, optimized for GPU architectures, provides significantly higher efficiency than existing methods. The proposed scheme offers a robust solution for real-time image encryption by demonstrating strong resistance to various types of attacks. The paper by Elrefaey *et al.* (2021) introduces a parallel implementation of a chaotic map-based image encryption algorithm using the Baker map and the Chen map. This parallel implementation, using GPUs, significantly speeds up the encryption and decryption processes for high-resolution images and long videos, making it suitable for real-time applications. The work by You *et al.* (2020) introduces a new algorithm based on hybrid chaotic maps for image encryption. By using a 1D logistic map and a 2D logistic chaotic dynamic system, the algorithm provides high security by scrambling pixel positions and values of images and accelerates the process using OpenCL. Song *et al.* (2022) proposes a fast and secure algorithm using intrinsic properties of chaotic systems, reversible steganography, and parallel computing for batch image encryption. The algorithm distributes batch images equally to each thread, applies Cipher Block Chaining (CBC) mode among neighboring images, and encrypts and embeds thread identifiers and CBC indexes into the encrypted images. By using the logistic map and the Chen system as chaotic systems in the encryption process, it enhances resistance to chosen-plaintext attacks. The paper by Bharadwaj *et al.* (2021) presents a GPU-accelerated implementation of an image encryption algorithm optimized with genetic algorithms. The algorithm provides high performance and security by using a modified XOR cipher to encrypt images and a genetic algorithm-based pseudo-random number generator with CUDA programming.

The main contributions of this paper to the literature are listed in order below:

- A new and simple encryption mechanism for three-dimensional images using the Logistic Map has been presented.
- The mobile computer version of the flagship GPU with CUDA support, which was released almost in the last year, has been used for the first time in the control center role and its performance has been tested.
- The importance of CUDA-supported encryption has been once again highlighted and proven with tests in the paper.
- The developed encryption technique has been implemented and the resulting outputs have been analyzed.

The remainder of this paper is organized as follows: Section 2 addresses the chaoticity analysis of the Logistic Map. Section 3 explains the generation of random numbers and randomness tests. Subsequently, Section 4 introduces the chaos-based encryption technique. Section 5 presents the analysis of images encrypted with this technique. Section 6 compares the proposed technique with other existing works. Finally, Section 7 discusses the conclusions and future work.

CHAOTIC ANALYSIS OF THE LOGISTIC MAP

Discrete-Time Logistic Map Chaotic System

In the recommended light encryption mechanism, the discrete-time chaotic system chosen for data encryption is the Logistic Map. The Logistic Map is particularly used to generate random numbers, which are crucial for encrypting image data through XOR operations. This chaotic system is preferred due to its ability to efficiently generate pseudo-random sequences necessary for the encryption process. The equation for the Logistic Map is presented in Equation 1. Here, X_n represents the current state, and r is the control parameter that dictates the system's behavior. By adjusting the value of r, the Logistic Map can produce a range of behaviors from steady-state to chaotic.

$$X_{n+1} = rX_n(1 - X_n) \tag{1}$$

The parameters necessary for the Logistic Map in this study are provided in Table 1. According to the table, the initial value $X(0)$ is set to 0.61, and the control parameter r is set to 3.9. For the random number generation process during encryption, 6,291,457 iterations were performed, and the least significant 4 bits of the binary representation of the number obtained in each iteration were used.

Time Series Analysis of the Values Generated by the Logistic Map

The time series analysis of the values generated by the Logistic Map is shown in Figure 1. The graph represents the values obtained from 250 iterations of the Logistic Map. The absence of periodicity in the time series demonstrates the chaotic behavior of the Logistic Map, which is essential for its effectiveness in encryption processes. This chaotic nature ensures that the values are unpredictable, thus enhancing the security of the encryption mechanism.

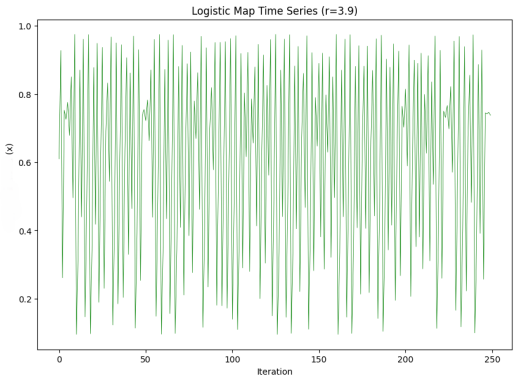


Figure 1 Time series of the values generated by the Logistic Map over 250 iterations. The plot illustrates the absence of periodicity, indicating the chaotic nature of the Logistic Map for $r = 3.9$.

Table 1 Logistic Map Parameters

Chaotic Systems	Logistic Map
X(0)	0.61
r value	3.9
Number of Iteration	6291457
Number of Binary	4

Initial Condition Sensitivity Analysis of the Logistic Map

The initial condition sensitivity analysis of the Logistic Map is depicted in Figure 2. The graph presents the time series of values generated over 250 iterations for two slightly different initial conditions: $X(0) = 0.61$ and $X(0) = 0.61 + 10^{-10}$. This analysis demonstrates that even a minute change in the initial condition results in significantly different trajectories, which is a hallmark of chaotic systems. The red and green lines represent the two initial conditions, respectively, and their rapid divergence emphasizes the sensitive dependence on initial conditions, underscoring the importance of precision in chaotic system applications.

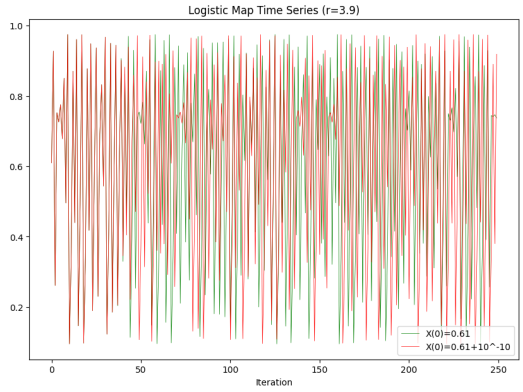


Figure 2 Time series of the Logistic Map values over 250 iterations for two slightly different initial conditions: $X(0) = 0.61$ and $X(0) = 0.61 + 10^{-10}$. The plot demonstrates the significant divergence in values due to the small change in the initial condition, illustrating the sensitive dependence on initial conditions characteristic of chaotic systems.

Logistic Map Chaotic System Function

The analysis utilizes a single map function due to the one-dimensional nature of the two nonlinear systems under consideration. The map functions are represented with X_n on the x-axis and X_{n+1} on the y-axis. Figure 3 illustrates the map function derived from data over 100,000 iterations. In this figure, there is a specific relationship between the values of X_n and X_{n+1}. These nonlinear systems demonstrate chaotic behavior as the values at each step differ from one another.

The graph in Figure 3 illustrates the relationship between X_n and X_{n+1} for the Logistic Map with $r = 3.9$. The plot shows how the population ratio changes iteratively, reflecting the quadratic nature of the Logistic Map. As the iterations proceed, the values generated by the map function demonstrate chaotic behavior, as evidenced by the lack of a repeating pattern and the sensitivity to initial conditions. This characteristic is essential for encryption processes, where unpredictability and complexity of the sequence are crucial for security.

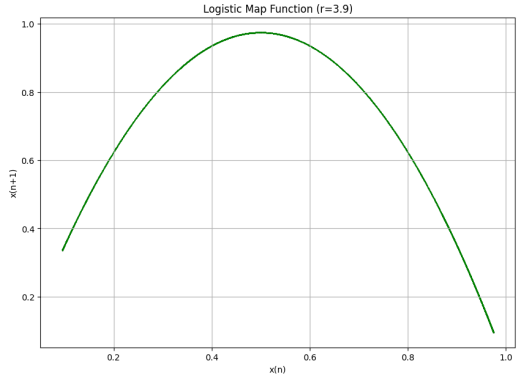


Figure 3 The map function of the Logistic Map for $r = 3.9$ over 100,000 steps. The x-axis represents X_n and the y-axis represents X_{n+1}.

Bifurcation Diagram of the Logistic Map

Bifurcation diagrams are a crucial tool for understanding the chaotic properties of a system. These diagrams illustrate the dynamic changes that occur as system parameters vary, allowing for an in-depth analysis of the system's chaotic behavior. Additionally, they help identify the parameter values at which the system exhibits chaotic or regular behavior.

For the Logistic Map system, the parameter r can range from 2 to 4, and the bifurcation diagram for this interval is presented in Figure 4. As depicted in the diagram, the system shows chaotic behavior within the range of 3.5 to 4.

RANDOM NUMBER GENERATION AND RANDOMNESS TESTS GENERATED BY THE LOGISTIC MAP

The Process of Generating Random Numbers

In this section, the process of generating random numbers used in the study is discussed. The Logistic Map chaotic system, introduced and analyzed for chaotic behavior in the previous section, will be utilized here.

The random number generator based on the Logistic Map is presented in Algorithm 1. Additionally, the parameters used in the algorithm are presented in Table 1. The algorithm starts with the parameter `r` and the initial condition `x0`. In the first step, the initial condition is used to calculate the first value with the Logistic Map

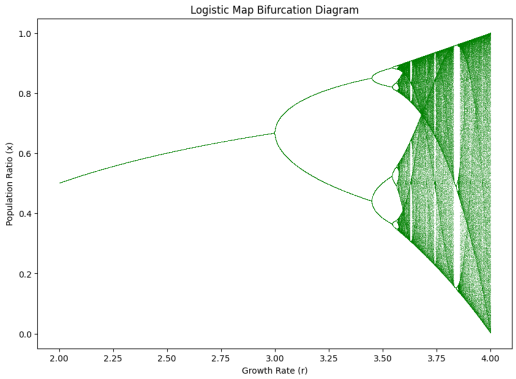


Figure 4 Bifurcation diagram of the Logistic Map for r values ranging from 2 to 4. The population ratio x is plotted against the growth rate r.

equation and assigned to `data[0]`. Then, for the specified number of steps, a loop is run where the Logistic Map equation is used in each step to calculate new values and add them to the list. The initial value is deleted from the list since it is used as the starting condition.

Subsequently, the number of binary digits to be obtained from each floating point number is specified, and these binary digits are added to the list called `binary_data_array`. This process continues by obtaining and adding the j-th lowest binary value of each floating point number to the list.

The obtained binary data list is divided into the specified number of parts (`num_parts`), and the size of each part (`part_size`) is calculated. Here, the number of parts is set to 8. These parts are assigned to the `parts` list, and each part is then added to the newly created `binary_data_array_new` list.

As a result, the random numbers generated using the Logistic Map are converted into binary values, which are then organized for later use. This algorithm creates an effective random number generator by leveraging the randomness characteristic of the Logistic Map chaotic system, thereby ensuring reliable random number generation.

Randomness Analysis of the Generated Random Numbers

In this subsection, the randomness analysis of the random numbers generated by the algorithm introduced in the previous subsection is conducted. The randomness analysis was performed using the NIST-800-22 and ENT (Beirami *et al.* 2012) tests.

The NIST-800-22 test, which is widely used worldwide, consists of 15 different statistical tests (Koyuncu 2014). For the test to be considered successful, all these statistical tests must pass. This provides evidence that the generated numbers are random. The NIST-800-22 test produces results based on the P-value, and for the results to be considered successful, the P-value must be greater than the threshold value of 0.001. In this study, the first 3,000,000 binary values generated by the Logistic Map were used. The results obtained are presented in Table 2. As seen in the table, all tests are considered successful as they meet the required conditions.

Another significant test for assessing the randomness in this study is the ENT test. Developed by John Walker, this test includes five different statistical evaluations. The entropy test measures the information density of the sequence, expressed as the number of bits per byte, with an ideal entropy level of 8. Entropy indicates the amount of information contained in the generated sequence. The chi-square test assesses the randomness of the sequence and

Algorithm 1 Pseudorandom number generation pseudo code

Result: Random Binary Numbers
Start
Enter parameter `r=3.9`
Enter initial condition `x0` $\leftarrow 0.1$
Number of steps `number_of_steps`
`data[0]` $\leftarrow$ `equation(x0)`
for $i \leftarrow 1$ to `number_of_steps` **do**
    `data[i]` $\leftarrow$ `equation(data[i-1])`
delete(`data[0]`)
`binary_data_array` $\leftarrow$ []
Binary number to get from float `number_of_binary`
for $i \leftarrow 0$ to `number_of_steps-1` **do**
    for $j \leftarrow 1$ to `number_of_binary` **do**
        Get j-th lowest value binary of `data[i]` $\rightarrow$ `binary_data`
        Append `binary_data` to `binary_data_array`
Number of parts to divide the list into
`num_parts` $\leftarrow 8$
`part_size` $\leftarrow$ `len(binary_data_array) // num_parts`
`parts` $\leftarrow$ `[binary_data_array[i*part_size:(i+1)*part_size]` for i in range(`num_parts`)]
Initialize the new list `binary_data_array_new` $\leftarrow$ []
for $i \leftarrow 0$ to `part_size-1` **do**
    for `part` in `parts` **do**
        Append `part[i]` to `binary_data_array_new`
End

is highly sensitive to errors in the PRNG. The arithmetic mean is computed by dividing the sum of all bytes in the generated file by the file length, with a value close to 127 indicating randomness. The Monte Carlo test demonstrates that the sequence converges to π with a minimal error of 0.01 percent, suggesting the sequence's proximity to randomness. The serial correlation coefficient test ensures that there is no correlation between each byte and the previous byte in the sequence. All generated binary random numbers were subjected to this test, and the results are presented in Table 3. As shown in the table, the results are close to the optimal values that define randomness. Therefore, the outcomes from both tests confirm that the numbers can be used reliably.

OVERVIEW OF THE NEW CHAOS-BASED ENCRYPTION TECHNIQUE

In this section, the encryption technique will be introduced. This technique will be used in the control center that controls the devices. The communication structure of the control center with other devices is shown in Figure 5. When the mobile control center communicates with other devices, the data is encrypted and decrypted with the support of the GPU. Thus, although the mobile control center has a weaker processing capacity compared to other advanced control centers, the delay in data encryption and decryption is minimized thanks to the CUDA support of the GPU.

In the study, encryption initially starts with the generation of random numbers as described in the previous section. The generated random numbers are then applied to RGB or Gray-Scale images to encrypt these images. The encryption of each channel is done separately as shown in Figure 6. The process of encrypting the random numbers for each of the RGB channels is presented in Algorithm 2. The first phase of the algorithm involves loading the data into the GPU and performing the XOR operation. Initially, the generated random numbers are split into three lists using

Table 2 Logistic Map P-Values and Test Results

Test	P-Value (Logistic Map)	Result
Frequency Monobit	0.60712	Passed
Frequency Test (M = 128)	0.03616	Passed
Run Test	0.72421	Passed
Test for The Longest	0.44117	Passed
Binary Matrix Rank	0.62683	Passed
Discreet Fourier Transform	0.84356	Passed
Non-overlapping (Single, B = 111 111 111)	0.00069	Passed
Overlapping Temp (B = 111 111 111)	0.07939	Passed
Maurier's Universal	0.20237	Passed
Linear Complexity (M = 500)	0.15630	Passed
Serial Test-1 (m = 16)	0.08933	Passed
Serial Test-2 (m = 16)	0.19621	Passed
Approximate Entropy (m = 10)	0.74849	Passed
Cumulative Sums Test (Forward)	0.57423	Passed
Random Excursion Test (x = -1)	0.62257	Passed
Random Excursion Variant Test (x = -6)	0.65029	Passed

Table 3 ENT Test Results

Test Name	Average	Ideal Results	Result
Arithmetic Mean	127.45217	127.5	Success
Entropy	7.99993	8	Success
Correlation	0.00032	0	Success
Chi Square	278.19856	10% and 90% between	Success
Monte Carlo	3.13186	3.1415	Success

the `split_list_into_three` function. Then, the CUDA kernel is defined, and the necessary kernel code for the XOR operation is created. This kernel code is a simple CUDA function that performs the XOR operation between two lists at each index. After the kernel code is compiled, preparations are made to run the function in the CUDA context. In this phase, the lists are converted to numpy arrays, and empty arrays are created to store the results. Subsequently, these numpy arrays are copied to the GPU memory. The block and grid sizes are defined, and the XOR operation is executed on the GPU. Finally, the calculated results are copied from the GPU memory to the CPU memory, completing the process. Thus, the encryption data for each channel is obtained.

The process of applying the encryption data generated for each channel to each channel is shown in Algorithm 3. The algorithm involves applying the encryption data to the RGB or Gray-Scale image channels. First, a new CUDA kernel code is defined, and each channel of the images is flattened. Then, these channels are divided into 64 parts, and each part is assigned as a new channel list. After creating empty arrays to store the results, the data is copied to the GPU memory. The block and grid sizes are defined, and the XOR operation is performed on the GPU. Once the process is complete, the calculated results are copied from the GPU

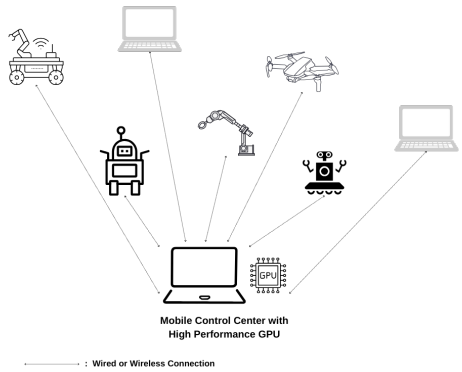


Figure 5 The wired or wireless connection structure of the mobile control center with other devices.

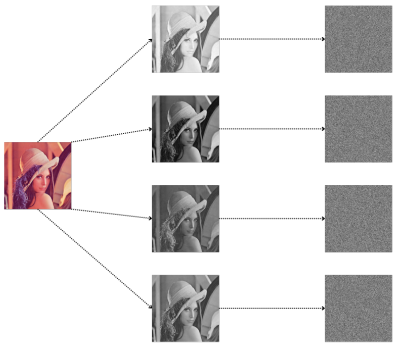


Figure 6 Encryption of each channel in RGB images or Gray-Scale image.

memory to the CPU memory. Finally, the results obtained for each channel are combined to produce the final encrypted image.

TESTS OF IMAGES ENCRYPTED WITH THE NEW TECHNIQUE

Image Analysis between Encrypted Image and Original Image

This section examines the differences between the original image and the encrypted image using the Lena image, a standard in image processing. Histogram, correlation, and entropy analyses were conducted on both the original and encrypted images, and differential attack analyses (NPCR and UACI) were performed on the encrypted image. NPCR evaluates the percentage of differing pixels between two encrypted images generated by altering a single pixel in the plaintext image, while UACI measures the average of the absolute differences between corresponding pixels in two encrypted images created by altering a single pixel in the plaintext image. In the NPCR test, two nearly identical plaintext images, differing by only one pixel, are encrypted, with a high NPCR value indicating strong resistance to differential attacks. Similarly, in the UACI test, two nearly identical plaintext images, differing by only one pixel, are encrypted, with a low UACI value indicating strong resistance to differential attacks. For the encryption method to be considered resistant to differential attacks, the NPCR value should be at least 99.9% and the UACI value should be less than 0.01. Additionally, to assess the performance of the new encryption method, histogram, correlation, and entropy analyses were carried

Algorithm 2 Encryption data for R (or gray-scale), G, B channels

Result: Encryption data for R (or gray-scale), G, B channels
Start
`part1, part2, part3 ← split_list_into_three(binary_data_array_new)`
Define CUDA kernel code for XOR operation:
`kernel_code ← "`
`__global__ void xor_lists(int *list1, int *list2, int *result, int size) {`
`int idx = threadIdx.x + blockIdx.x * blockDim.x;`
`if (idx < size) {`
`result[idx] = list1[idx] ^ list2[idx]; } }"`
`mod ← SourceModule(kernel_code)`
`xor_lists ← mod.get_function("xor_lists")`
Convert lists to numpy arrays
`part1_np ← part1, part2_np ← part2, part3_np ← part3`
Allocate space for results
`zeros_like(r_kanal_list←part1_np, g_kanal_list←part2_np, b_kanal_list←part3_np)`
Copy lists to GPU memory
`cuda.mem_alloc(part1_gpu←part1_np, part2_gpu←part2_np, part3_gpu←part3_np, r_kanal_gpu←r_kanal_list, g_kanal_gpu←g_kanal_list, b_kanal_gpu←b_kanal_list)`
`cuda.memcpy_htod(part1_gpu←part1_np, part2_gpu←part2_np, part3_gpu←part3_np)`
Define block and grid sizes
`block_size ← 256`
`grid_size ← ceil(len(part1_np) / block_size)`
Perform XOR operation on GPU
`xor_lists(part1_gpu, part2_gpu, r_kanal_gpu, len(part1_np), block=(block_size, 1, 1), grid=(grid_size, 1))`
`xor_lists(part1_gpu, part3_gpu, g_kanal_gpu, len(part1_np), block=(block_size, 1, 1), grid=(grid_size, 1))`
`xor_lists(part2_gpu, part3_gpu, b_kanal_gpu, len(part2_np), block=(block_size, 1, 1), grid=(grid_size, 1))`
Copy results back to CPU
`cuda.memcpy_dtoh(r_kanal_list←r_kanal_gpu, g_kanal_list←g_kanal_gpu, b_kanal_list←b_kanal_gpu)`
End

out on both the original and encrypted images. The histogram displays the distribution of pixel intensities in an image. The formula for calculating the histogram of an image is:

$$H(i) = \sum_{j=0}^{L-1} \delta(i - f(j)) \tag{2}$$

Correlation measures the linear relationship between two variables. The formula for calculating correlation is:

$$C = \frac{\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}[(f(i,j)-\overline{f})(g(i,j)-\overline{g})]}{\sqrt{\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}(f(i,j)-\overline{f})^2 \sum_{i=0}^{N-1}\sum_{j=0}^{N-1}(g(i,j)-\overline{g})^2}} \tag{3}$$

Entropy measures the amount of information in an image and expresses the degree of randomness or uncertainty in the data input. High entropy indicates that the data is more unpredictable

Algorithm 3 CUDA based XOR Operation for Any Channel

```
Result: Encrypted Image Channel
Start
Define new CUDA kernel code for image XOR operation (The definition is the same as in the Algorithm 2.)
Load channel images and flatten them
load_image_as_array(any_channel ← any_channel_file)
Split the list into 64 parts
num_parts ← 64
part_size ← len(any_kanal_list) // num_parts
Assign the first part as new channel lists
any_kanal_new_list ← any_kanal_list[:part_size]
Allocate space for results
zeros_like(any_result ← any_kanal_new_list)
Copy data to the GPU
cuda.mem_alloc(any_channel_gpu ← any_channel)
cuda.mem_alloc(any_list_gpu ← any_kanal_new_list.nbytes, any_result_gpu ← any_result)
cuda.memcpy_htod(any_channel_gpu ← any_channel, any_list_gpu ← any_kanal_new_list)
Define grid and block sizes
block_size ← 256
grid_size ← ceil(any_channel.size / block_size)
Perform XOR operation on GPU
xor_lists(any_channel_gpu, any_list_gpu, any_result_gpu, any_channel.size, block=(block_size, 1, 1), grid=(grid_size, 1))
Copy results back to CPU
cuda.memcpy_dtoh(any_result ← any_result_gpu)
Combine channels back into images
image_shape ← (image_size_1, image_size_2)
Process Any channel result
any_result_image ← any_result.reshape(image_shape)
End
```

and thus harder to compress or encrypt. In encryption, high entropy means more randomness in the input, which makes it harder for an attacker to detect patterns or weaknesses in the encryption algorithm. Low entropy indicates that there may be too many patterns or repetitions in the data, which can make it easier for an attacker to decrypt the message. Therefore, to ensure strong encryption, a balance between entropy and predictability must be established. The formula for calculating entropy is:

$$S = -\sum_{i=1}^{N} p_i \log_2(p_i) \tag{4}$$

Figure 7 displays the histogram analyses of the original image (shown on the left) in Figure 6, as well as its encrypted versions in the order of R, G, B, and grayscale channels. In Figure 7, the R, G, B, and grayscale channels are presented separately as histogram plots. Examining the histogram plots of the different channels of the original image, it is observable that each channel contains distinct information. However, when analyzing the histogram plots of images encrypted using new random numbers generated by the Logistic Map, these plots can be considered as purely noise data. This indicates that no meaningful information can be extracted from the histogram of the image.

Figure 8 presents the cross-correlation plots for the R, G, B, and gray-scale channels of the same image. As can be seen, while information can be extracted from the channels of the original image, the situation in the encrypted image channels appears spread out and homogeneous. This indicates that the image channels have been well encrypted.

Tables 4 and 5 provide the Horizontal and Vertical Correlation Coefficient values for the original image and its encrypted versions. Calculations have been performed separately for each channel (R, G, B, grayscale). As seen in the tables, for the original image in different dimensions, both horizontal and vertical correlation values for each channel range from 0.7 to 1.0, indicating the presence of information within the image. However, after encrypting the image channels using random numbers generated from the Logistic Map in different dimensions, these values have approached 0. This demonstrates that the encryption is robust.

Table 6 presents the NPCR, UACI, and Entropy values obtained from different sizes and channels of the Lena image after encryption (including the entropy values of the original images). Here, the NPCR value shows the normalized form of the changing values for 8-bit pixels between 0 and 1, while the UACI values indicate the normalized form of the intensity of changing pixels, also between 0 and 1. Considering all these values, it is evident that the discrete-time chaotic encryption technique is successful.

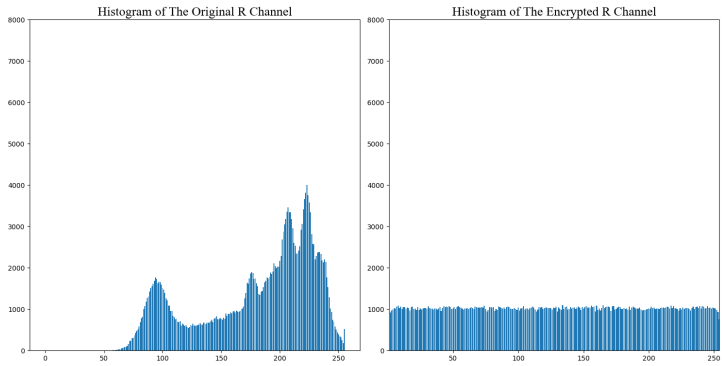
Processing Time of the Encryption Technique in Mobile Control Center for Video-Based Applications

In this sub-section, the encryption speeds of video frames from devices connected to the Mobile Control Center will be analyzed. Images with sizes of 128, 256, 512, and 1024 were used for encryption. In the simulation environment, the number of connected devices was set to 1, 5, 10, 20, 50, and 100. Additionally, the laptop version of the NVIDIA GeForce RTX 4090 model was used for encryption. The average number of frames encrypted by the control center from the image data received from each device was calculated. The results obtained are presented in Table 7. The table shows the number of frames processed according to image sizes and the number of devices.

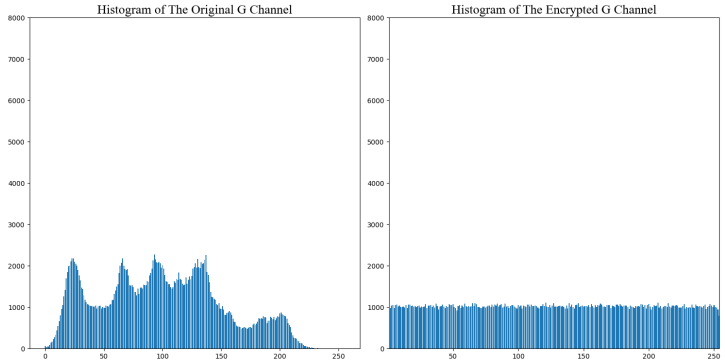
When examining the obtained results, it is observed that the significant drop in fps occurs with the increase in image sizes. Although the increase in the number of devices for each size decreases the fps value, this decrease is not significant. Particularly, the GPU used for encryption has not been heavily burdened during these processes. There has only been a busy state in CPU-GPU read/write operations. However, this busy state is not at a level that would affect other tasks. This indicates that the proposed technique is efficient.

COMPARISON OF THE PROPOSED TECHNIQUE WITH EXISTING WORKS

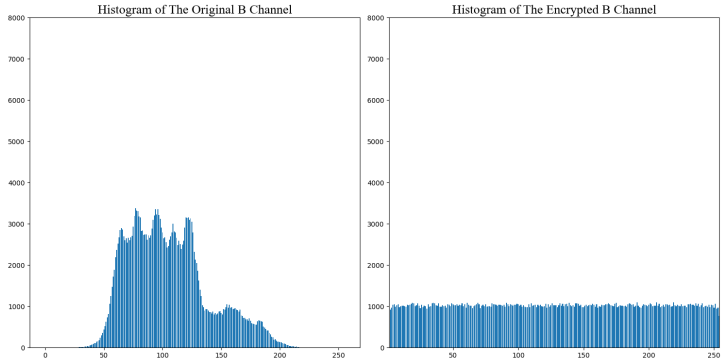
In this section, the performance of the Lena image encrypted with the proposed encryption technique is compared with that of the Lena image encrypted by other studies. Generally, recent and significant studies have been used for this comparison. The grayscale version of the Lena image was used, focusing on correlation coefficients, entropy, and NPCR values. The obtained test results are presented in Table 8. In the table, while the proposed study achieved very good results compared to some other studies, it obtained values very close to those of the studies with the best results. This demonstrates the effectiveness of the encryption technique.



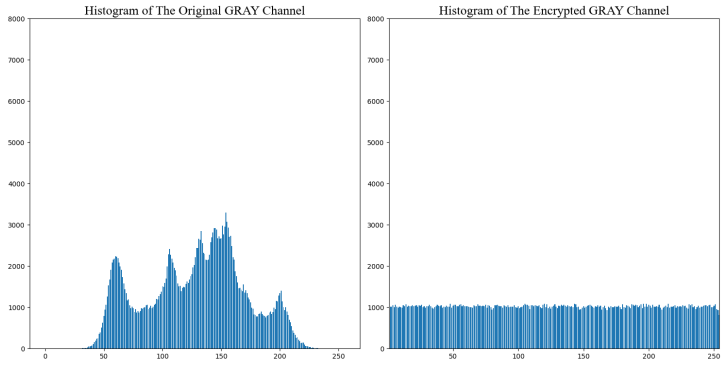
(a) Histogram of The Original and Encrypted R Channel



(b) Histogram of The Original and Encrypted G Channel

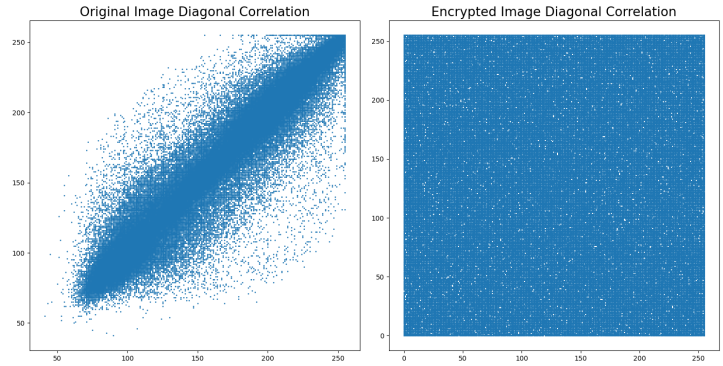


(c) Histogram of The Original and Encrypted B Channel

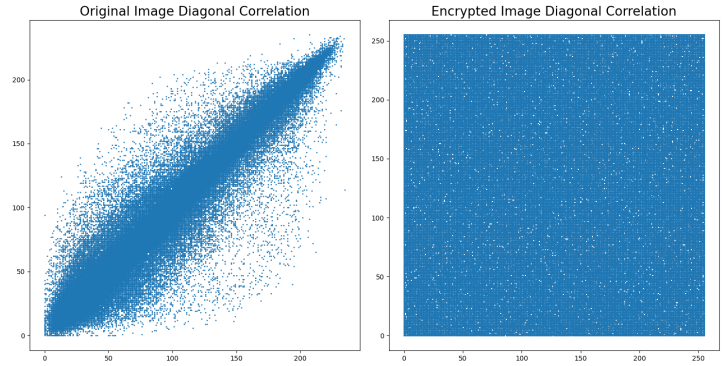


(d) Histogram of The Original and Encrypted Gray Channel

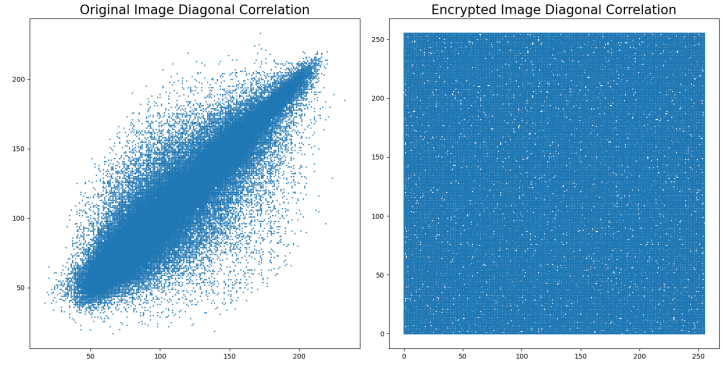
Figure 7 Comparative Histogram Analysis of Original and Encrypted Image Channels



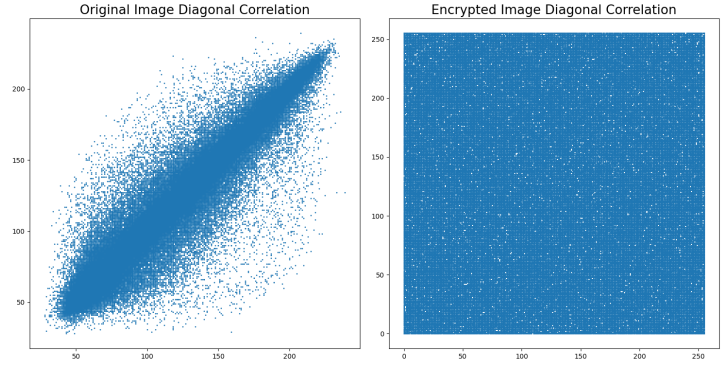
(a) Diagonal Correlation of The Original and Encrypted R Channel



(b) Diagonal Correlation of The Original and Encrypted G Channel



(c) Diagonal Correlation of The Original and Encrypted B Channel



(d) Diagonal Correlation of The Original and Encrypted Gray Channel

Figure 8 Comparative Diagonal Correlation Analysis of Original and Encrypted Image Channels

Table 4 Logistic Map Horizontal Correlation

Size	Channel	Horizontal Corr. (Original)	Horizontal Corr. (Encrypted)
128x128	Red	0.9539	0.0053
128x128	Green	0.9581	0.0074
128x128	Blue	0.9191	-0.0046
128x128	Gray-Scale	0.9442	0.0040
256x256	Red	0.9742	-0.0001
256x256	Green	0.9768	0.0072
256x256	Blue	0.9515	-0.0045
256x256	Gray-Scale	0.9684	0.0002
512x512	Red	0.9888	-0.0005
512x512	Green	0.9899	-0.0014
512x512	Blue	0.9783	-0.0001
512x512	Gray-Scale	0.9864	-0.0011

Table 5 Logistic Map Vertical Correlation

Size	Channel	Vertical Corr. (Original)	Vertical Corr. (Encrypted)
128x128	Red	0.8654	0.0005
128x128	Green	0.8463	0.0136
128x128	Blue	0.7802	-0.0145
128x128	Gray-Scale	0.8280	0.0064
256x256	Red	0.9255	-0.0043
256x256	Green	0.9144	0.0016
256x256	Blue	0.8736	0.0017
256x256	Gray-Scale	0.9035	0.0003
512x512	Red	0.9676	-0.0005
512x512	Green	0.9630	-0.0011
512x512	Blue	0.9448	-0.0010
512x512	Gray-Scale	0.9580	-0.0011

Table 6 Logistic Map NPCR, UACI, Entropy

Size	Channel	NPCR	UACI	Entropy (Original)	Entropy (Encrypted)
128x128	Red	0.9960	0.3246	7.2996	7.9875
128x128	Green	0.9959	0.3031	7.5558	7.9861
128x128	Blue	0.9961	0.2786	7.1104	7.9854
128x128	Gray-Scale	0.9967	0.2753	7.3255	7.9892
256x256	Red	0.9962	0.3296	7.2835	7.9961
256x256	Green	0.9958	0.3036	7.5801	7.9963
256x256	Blue	0.9962	0.2765	7.0603	7.9957
256x256	Gray-Scale	0.9958	0.2787	7.3176	7.9961
512x512	Red	0.9962	0.3309	7.2709	7.9981
512x512	Green	0.9961	0.3050	7.5860	7.9981
512x512	Blue	0.9958	0.2761	7.0022	7.9979
512x512	Gray-Scale	0.9960	0.2796	7.3015	7.9981

Table 7 FPS Test Results

Size	Devices	fps	Size	Devices	fps
128x128	1	3456	512x512	1	376
128x128	5	3280	512x512	5	360
128x128	10	2900	512x512	10	316
128x128	20	2404	512x512	20	284
128x128	50	1960	512x512	50	232
128x128	100	1284	512x512	100	168
256x256	1	1484	1024x1024	1	148
256x256	5	1424	1024x1024	5	144
256x256	10	1240	1024x1024	10	132
256x256	20	924	1024x1024	20	116
256x256	50	644	1024x1024	50	104
256x256	100	264	1024x1024	100	76

Table 8 Comparison of Studies

Studies	Correlation Coefficient				Entropy	NPCR (%)
	256		512		512	512
	Horizontal	Vertical	Horizontal	Vertical		
Shakir (2019)	N/A	N/A	0.1394	0.0339	7.7362	N/A
Kiran *et al.* (2023)	0.0022	0.0027	-0.0016	-0.0025	7.9981	99.606
You *et al.* (2020)	N/A	N/A	0.0149	0.0151	7.9457	99.65
Clemente-López *et al.* (2024)	N/A	N/A	0.0049	0.0038	7.9992	99.61
Our Study	0.0002	0.0003	-0.0011	-0.0011	7.9981	99.60

CONCLUSION

This study presents a novel chaos-based encryption technique leveraging the parallel processing capabilities of CUDA for mobile control centers with powerful GPUs. The proposed method utilizes the Logistic Map for random number generation, which is then applied to the RGB and grayscale channels of image data through XOR operations. The results of our experiments demonstrate that the technique provides fast encryption speeds and high security levels, making it suitable for real-time applications in IoT environments. The statistical analyses, including histogram, correlation, and entropy tests, confirm the robustness and effectiveness of the encryption mechanism against various types of attacks.

Future work will focus on several enhancements and extensions of the current study. One area of improvement involves optimizing the algorithm for different hardware configurations, including low-power devices, to ensure broader applicability. Additionally, integrating more sophisticated chaotic maps and exploring hybrid encryption techniques could further enhance security and performance. We also plan to conduct extensive testing in real-world IoT scenarios to validate the practicality and reliability of the proposed method. Furthermore, expanding the scope of the study to include other types of data, such as video and sensor data, will help in assessing the versatility and scalability of the encryption technique.

Availability of data and material

Not applicable.

Conflicts of interest

The author declares that there is no conflict of interest regarding the publication of this paper.

Ethical standard

The author has no relevant financial or non-financial interests to disclose.

LITERATURE CITED

Beirami, A., H. Nejati, and W. Ali, 2012 Zigzag map: a variability-aware discrete-time chaotic-map truly random number generator. Electronics Letters **48**: 1537–1538.

Bezerra, J. I. M., A. Molter, G. Machado, R. I. Soares, and V. V. d. A. Camargo, 2024 A novel single kernel parallel image encryption scheme based on a chaotic map. Journal of Real-Time Image Processing **21**: 1–13.

Bharadwaj, B., J. Saira Banu, M. Madiajagan, M. R. Ghalib, O. Castillo, *et al.*, 2021 Gpu-accelerated implementation of a genetically optimized image encryption algorithm. Soft Computing **25**: 14413–14428.

Boyraz, O. F., E. Guleryuz, A. Akgul, M. Z. Yildiz, H. E. Kiran, *et al.*, 2022 A novel security and authentication method for infrared medical image with discrete time chaotic systems. Optik **267**: 169717.

Clemente-Lopez, D., J. de Jesus Rangel-Magdaleno, and J. M. Muñoz-Pacheco, 2024 A lightweight chaos-based encryption scheme for iot healthcare systems. Internet of Things **25**: 101032.

Clemente-López, D., J. M. Munoz-Pacheco, and J. de Jesus Rangel-Magdaleno, 2024 Experimental validation of iot image encryption scheme based on a 5-d fractional hyperchaotic system and numba jit compiler. Internet of Things **25**: 101116.

Elrefaey, A., A. Sarhan, and N. M. El-Shennawy, 2021 Parallel approaches to improve the speed of chaotic-maps-based encryption using gpu. Journal of Real-Time Image Processing pp. 1–10.

Erkan, E., H. Oğraş, and Ş. Fidan, 2023 Application of a secure data transmission with an effective timing algorithm based on lora modulation and chaos. Microprocessors and Microsystems **99**: 104829.

Francisti, J., Z. Balogh, J. Reichel, M. Magdin, Š. Koprda, *et al.*, 2020 Application experiences using iot devices in education. Applied Sciences **10**: 7286.

Ghosh, G., f. Kavita, D. Anand, S. Verma, D. B. Rawat, *et al.*, 2021 Secure surveillance systems using partial-regeneration-based non-dominated optimization and 5d-chaotic map. Symmetry **13**: 1447.

Jadhav, S., U. Patel, A. Natu, B. Patil, and S. Palwe, 2023 Cryptography using gpgpu. In *Intelligent Communication Technologies and Virtual Mobile Networks*, pp. 299–313, Springer.

Kashyap, P. K., S. Kumar, A. Jaiswal, M. Prasad, and A. H. Gandomi, 2021 Towards precision agriculture: Iot-enabled intelligent irrigation systems using deep learning neural network. IEEE Sensors Journal **21**: 17479–17491.

Kiran, H. E., A. Akgul, O. Yildiz, and E. Deniz, 2023 Lightweight encryption mechanism with discrete-time chaotic maps for internet of robotic things. Integration **93**: 102047.

Koyuncu, İ., 2014 *Kriptolojik uygulamalar için FPGA tabanlı yeni kaotik osilatörlerin ve gerçek rasgele sayı üreteçlerinin tasarımı ve*

gerçeklenmesi. Ph.D. thesis, Sakarya Universitesi (Turkey).
Kumari, P. and B. Mondal, 2023 An encryption scheme based on grain stream cipher and chaos for privacy protection of image data on iot network. Wireless Personal Communications **130**: 2261–2280.
Mai, X. H., B. Zhang, X. S. Luo, *et al.*, 2015 Controlling chaos in complex motor networks by environment. IEEE Transactions on Circuits and Systems II: Express Briefs **62**: 603–607.
Man, Z., J. Li, X. Di, X. Liu, J. Zhou, *et al.*, 2021 A novel image encryption algorithm based on least squares generative adversarial network random number generator. Multimedia Tools and Applications **80**: 27445–27469.
Naik, R. B. and U. Singh, 2024 A review on applications of chaotic maps in pseudo-random number generators and encryption. Annals of Data Science **11**: 25–50.
Rajendran, S. and M. Doraipandian, 2021 Chaos based secure medical image transmission model for iot-powered healthcare systems. In *IOP Conference Series: Materials Science and Engineering*, volume 1022, p. 012106, IOP Publishing.
Romeo, L., A. Petitti, R. Marani, and A. Milella, 2020 Internet of robotic things in smart domains: Applications and challenges. Sensors **20**: 3355.
Shakir, H. R., 2019 An image encryption method based on selective aes coding of wavelet transform and chaotic pixel shuffling. Multimedia Tools and Applications **78**: 26073–26087.
Singh, R., A. Gehlot, S. V. Akram, L. R. Gupta, M. K. Jena, *et al.*, 2021 Cloud manufacturing, internet of things-assisted manufacturing and 3d printing technology: reliable tools for sustainable construction. Sustainability **13**: 7327.
Song, W., C. Fu, M. Tie, C.-W. Sham, J. Liu, *et al.*, 2022 A fast parallel batch image encryption algorithm using intrinsic properties of chaos. Signal Processing: Image Communication **102**: 116628.
Tutueva, A. V., E. G. Nepomuceno, A. I. Karimov, V. S. Andreev, and D. N. Butusov, 2020 Adaptive chaotic maps and their application to pseudo-random numbers generation. Chaos, Solitons & Fractals **133**: 109615.
You, L., E. Yang, and G. Wang, 2020 A novel parallel image encryption algorithm based on hybrid chaotic maps with opencl implementation. Soft Computing **24**: 12413–12427.

How to cite this article: Kiran, H. E. A Novel Chaos-Based Encryption Technique with Parallel Processing Using CUDA for Mobile Powerful GPU Control Center. *Chaos and Fractals*, 1(1), 6-18, 2024.

ADBA Chaos and Fractals
Cutting-Edge Scientific Solutions

e-ISSN: 3023-8951
RESEARCH ARTICLE
Vol.1 / No.1 / 2024 / pp.19-30
https://doi.org/10.69882/adba.chf.2024073

Comparative Analysis of State-of-the-Art Q&A Models: BERT, RoBERTa, DistilBERT, and ALBERT on SQuAD v2 Dataset

Cem Özkurt *,[1]
*Sakarya University of Applied Sciences, Artificial Intelligence and Data Science Application and Research Center, 54050, Sakarya, Türkiye.

ABSTRACT In the rapidly evolving landscape of natural language processing (NLP) and artificial intelligence, recent years have witnessed significant advancements, particularly in text-based question-answering (QA) systems. The Stanford Question Answering Dataset (SQuAD v2) has emerged as a prominent benchmark, offering diverse language understanding challenges. This study conducts a thorough examination of cutting-edge QA models—BERT, DistilBERT, RoBERTa, and ALBERT—each featuring distinct architectures, focusing on their training and performance on SQuAD v2. The analysis aims to uncover the unique strengths of each model, providing insights into their capabilities and exploring the impact of different training techniques on their performance. The primary objective is to enhance our understanding of text-based QA systems' evolution and their effectiveness in real-world scenarios. The results of this comparative study are poised to influence the utilization and development of these models in both industry and research. The investigation meticulously evaluates BERT, ALBERT, RoBERTa, and DistilBERT QA models using the SQuAD v2 dataset, emphasizing instances of accurate responses and identifying areas where completeness may be lacking. This nuanced exploration contributes to the ongoing discourse on the advancement of text-based question-answering systems, shedding light on the strengths and limitations of each QA model. Based on the results obtained, ALBERT achieved an exact match of 86.85% and an F1 score of 89.91% on the SQuAD v2 dataset, demonstrating superior performance in both answerable ('HasAns') and unanswerable ('NoAns') questions. BERT and RoBERTa also showed strong performance, while DistilBERT lagged slightly behind. This study provides a significant contribution to the advancement of text-based question-answering systems, offering insights that can shape the utilization of these models in both industry and research domains.

KEYWORDS
Question-Answering models
BERT
RoBERTa
DistilBERT
ALBERT
SQuAD v2 Dataset
Model performance

INTRODUCTION

Recent years have witnessed significant advancements in natural language processing (NLP) and artificial intelligence, particularly in text-based question-answering (QA) systems. Research in this field aims to develop new methods and approaches to enhance the effectiveness of QA systems used in various domains. These developments have led to the emergence of benchmarks such as the Stanford Question Answering Dataset (SQuAD v2) and facilitated the comparison of QA model performances. For instance, one such study introduces "RealTime QA," a dynamic QA platform that poses questions about current world events and regularly evaluates systems. This platform challenges the static assumptions present in traditional open-domain QA datasets and emphasizes real-time applications. The models developed serve as strong baseline models built upon large pretrained language models, thus making significant strides in real-time QA services (Kasai *et al.* 2024).

Additionally, a system named "Visconde" proposes a solution for answering questions that require evidence spread across multiple documents. This system employs a three-step pipeline to address the task, highlighting that current retrievers are often the primary bottleneck and that models perform at human levels given relevant passages (Pereira *et al.* 2023).

Furthermore, a review on how transformer models are applied in text-based QA systems sheds light on recent trends in the field. This review discusses different transformer architectures, attention mechanisms, and evaluation metrics used to assess the performance of QA systems (Nassiri and Akhloufi 2023). Moreover, a study addressing the information-intensive construction industry

Manuscript received: 25 June 2024,
Revised: 10 July 2024,
Accepted: 11 July 2024.

[1]cemozkurt@subu.edu.tr (**Corresponding author**)

develops a query-answering (QA) system using natural language processing (NLP) methods. This system aims to support decision-making processes in construction projects by creating virtual assistants (Wang *et al.* 2022). The contributions of these studies not only aid in understanding and improving current text-based QA applications but also lay a foundation for future research endeavors. These advancements have the potential to make QA systems more effective and usable in both industry and academia (Caballero 2021).

In recent years, Natural Language Processing (NLP) has undergone remarkable advancements, primarily attributed to the transformative influence of the Transformer architecture and the selfattention mechanism. These attention-based models have showcased unparalleled performance across various NLP benchmarks, fueled in part by the growing popularity of transfer learning. This section provides an overview of related works that delve into the application and comparative analysis of Transformerbased models across different domains.

Rawat and Samant (2022) conducted a comparative analysis of transformer-based models for question answering, delving into models such as BERT, ALBERT, RoBERTa, XLNET, DistilBERT, Electra, and Pegasus. Their study, centered on Question-Answering (QA) systems using the SQUAD2 dataset, emphasizes the evolution from traditional "Bag of Words" methods to more efficient transformer libraries, exemplified by HuggingFace's BERT Question Answering model. This approach significantly enhances the models' capability to answer questions from large documents. Nassiri and Akhloufi (2023) contribute with a comprehensive review of studies focusing on the use of transformer models in text-based question-answering systems. The paper categorizes transformer architectures based on encoders, decoders, and encoder-decoder structures. It explores recent trends in textual QA datasets, providing insights into QA system architectures and evaluation metrics. The authors underscore the need for simplified implementation of Transformer models.

Kumari *et al.* (2022b) present a comparative analysis of transformer-tased models for document visual question answering, concentrating on Visual Question Answering (VQA), specifically the DocVQA task. The study investigates transformer models such as BERT, ALBERT, RoBERTa, ELECTRA, and Distil-BERT. The analysis includes a detailed examination of validation accuracy, considering challenges posed by documents, layout understanding, and writing patterns. Sabharwal and Agrawal (2021) explore the intricacies of the BERT algorithm for sentence embedding and various training strategies, providing a practical application in text classification systems. This chapter serves as a valuable resource for understanding BERT's applications in Neural Networks and Natural Language Processing.

Gillioz *et al.* (2020) offer an overview of transformer-based models for NLP tasks, discussing the transformative impact of the transformer architecture on NLP since its proposal in 2017. The authors cover auto-regressive models like GPT, GPT-2, and XLNET, as well as auto-encoder architectures like BERT and post-BERT models, including RoBERTa, ALBERT, ERNIE 1.0/2.0. Sidorov *et al.* (2023) analyze the performance of different transformer models for regret and hope speech detection, highlighting their effectiveness and superiority in regret detection. The study emphasizes the importance of considering specific transformer architectures and pre-training for different tasks.

Pirozelli *et al.* (2022) propose an innovative approach to QA systems, exploring dual system architectures that filter unanswerable or meaningless questions. The paper presents experiments using classification and regression models to filter questions, demonstrating that this modular approach contributes to improving the quality of answers generated by QA systems. Nassiri and Akhloufi (2023) delve into the application of transformer models in text-based questionanswering systems, emphasizing their significance in natural language processing (NLP). The study provides a comprehensive review, categorizing transformer architectures based on encoders, decoders, and encoder-decoder structures. The authors also highlight recent trends in textual QA datasets, 2 system architectures, and evaluation metrics, underscoring the need for simplified implementation of transformer models.

Ghanem *et al.* (2023) tackle the issue of spam on social networks by proposing a RoBERTabased bi-directional Recurrent Neural Network for spam detection. Their study demonstrates superior performance, outperforming common transformer-based models on benchmark datasets from Twitter, YouTube, and SMS. MacRae (2022) details the development and deployment of NOLEdge, an intelligent search tool for the Florida State University Computer Science department. The study involves fine-tuning a pretrained transformer model and explores various methods of textual data augmentation, contributing insights into the model's efficacy and potential areas for further research.

Tahsin Mayeesha *et al.* (2021) address the gap in Bengali language processing, focusing on training transformer models for question answering. The study utilizes synthetic reading comprehension datasets and human-annotated Bengali QA datasets, comparing the models with human performance through survey experiments. Schütz *et al.* (2021) propose a content-based classification approach for automatic fake news detection using various pre-trained transformer models. The study reveals the effectiveness of transformers in achieving high accuracy on the FakeNewsNet dataset, emphasizing their potential impact on combating misinformation.

Kumari *et al.* (2022a) contribute to the field of question answering and generation, introducing novel transformer-based models like BERT, AIBERT, and DistilBERT. Their work integrates question generation with question answering systems, showcasing the models' capabilities in suggesting relevant questions based on input context. David (2020) explores the representation learning of autoencoding transformer models in ad hoc information retrieval, evaluating various transformer architectures such as BERT, RoBERTa, and DistilBERT. The study provides insights into the performance of these models in tasks like semantic similarity and their suitability for ad hoc document retrieval.

Malla and Alphonse (2021) address the COVID-19 outbreak, developing an ensemble pre-trained deep learning model for detecting informative tweets. Their Majority Voting technique-based Ensemble Deep Learning (MVEDL) model demonstrates high accuracy in identifying COVID-19 related informative tweets. Srivastava *et al.* (2021) investigate brand perception in marketing, probing contextual language models, including BERT and GPT, for associations with brand attributes. The study aims to understand the encoded brand perceptions and their potential impact on downstream tasks. Greco *et al.* (2022) provide a comprehensive comparison of transformer-based language models on NLP benchmarks, shedding light on the strengths and weaknesses of various models in different NLP tasks.

Sundelin (2023) explores the use of transformer models in identifying toxic language online, comparing the performance of RoBERTa, ALBERT, and DistilBERT. The study reveals distinctions in their efficiency based on datasets and real-world evaluations. Kumar *et al.* (2023) investigate modern question-answering ma-

chines, focusing on BERT and its variants. The study outlines the operation of machine reading comprehension and its application in providing in-depth solutions to user queries.

In conclusion, these collective works significantly contribute to the understanding of transformer models' applications across diverse domains, showcasing their effectiveness in addressing complex challenges in natural language processing and information retrieval.

MATERIALS AND METHODS

One of the significant strengths of this study lies in its comprehensive comparative analysis of four different BERT-based models (DistilBERT, BERT, ALBERT, and RoBERTa) on the SQuAD v2 dataset. Unlike many other studies that focus on a single model or dataset, this research provides a detailed evaluation across multiple models, offering a broader perspective on their relative performances. Additionally, our work uniquely emphasizes the practical implications of model performance in specific scenarios. By considering computational resource limitations, we highlight DistilBERT's value, making this study highly relevant for applications with constrained resources. The inclusion of a thorough discussion on hyperparameter optimization and its impact on model performance further distinguishes our research. This attention to fine-tuning details provides actionable insights for practitioners looking to maximize the efficiency of BERT-based models. Furthermore, our analysis includes both answerable and unanswerable questions, offering a more nuanced understanding of each model's strengths and weaknesses. This dual focus enhances the study's applicability in real-world settings, where the ability to handle unanswerable questions is critical. Overall, the comprehensive approach, practical relevance, and detailed methodological insights contribute to making this study a valuable resource for the NLP community.

BERT and Its Impact

The advent of Bidirectional Encoder Representations from Transformers (BERT) by Devlin *et al.* (2018) has revolutionized the field of Natural Language Processing (NLP). BERT is a deep learning model designed to understand the context of a word in search queries, making it highly effective for tasks like question answering, language inference, and others.

BERT's transformer architecture allows it to achieve state-of-the-art results by pre-training on a large corpus and fine-tuning on specific tasks. The model employs a bidirectional approach, considering the context from both directions, which sets it apart from traditional models like LSTM and GRU. Studies by Liu *et al.* (2019); Yang *et al.* (2019) have shown BERT's superior performance in various NLP benchmarks.

Application in Question Answering

The SQuAD (Stanford Question Answering Dataset) has been a benchmark for evaluating question-answering models. The release of SQuAD v2 introduced unanswerable questions, adding a new challenge for models to determine when no answer is possible. Prior works such as by Lan *et al.* (2019) have addressed these challenges by enhancing model architectures and training strategies.

Our study builds on these foundations, aiming to fine-tune BERT on the SQuAD v2 dataset to push the boundaries of question-answering capabilities. By focusing on the nuances of unanswerable questions and optimizing hyperparameters, we seek to achieve a new benchmark in performance.

Datasets

For the training and evaluation of our question-answering systems, we employed the widely recognized "Squad v2" dataset, developed by Stanford University specifically for natural language processing (NLP) tasks. Serving as an enhanced iteration of "Squad v1," this dataset comprises over 100,000 instances, each consisting of a paragraph and a corresponding question.

The "Squad v2" dataset Yu and Sun (2023) is partitioned into two subsets: the training set, featuring 130,319 question-answer pairs, and the validation set, with 11,873 pairs. A key attribute of the dataset is the equitable distribution of answerable and unanswerable questions in both sets, ensuring a balanced evaluation approach. Stored in JSON format, each dataset file includes an array of data items. These items consist of a title and a set of paragraphs, wherein each paragraph contains text and a list of associated questions. Each question is uniquely identified by an ID, question text, answerability label, and one or more answers, each specified by text and character position. To quantify the linguistic complexity, the text in the "Squad v2" dataset undergoes tokenization, resulting in a total of 1,535,809 tokens. The training set encompasses 1,321,104 tokens, while the validation set comprises 214,705 tokens. Notably, the tokenization method segments text based on word boundaries and punctuation marks, eliminating spaces.

A distinctive feature of "Squad v2" is the incorporation of "impossible" instances, strategically designed to enhance accuracy measurement by introducing challenging scenarios. These instances include situations where answers are not present in the text or where questions contain incomplete or misleading information. The dataset's comprehensive coverage of diverse topics and real-world meaning contributes to the improvement of model generalization capabilities. Through the strategic utilization of the "Squad v2" dataset, this study aims to provide a robust and comprehensive performance measurement grounded in real-world scenarios, both during the training phase and result assessment.

Within Figure 1, we encounter a screenshot showcasing a question and its corresponding answer from the SQuAD v2 dataset. This dataset is a widely utilized question-answer dataset in natural language processing (NLP) research, composed of text and associated queries. The presented question in Figure 1 is, 'How did Marco Polo acquire information about China?' This question has sparked numerous debates due to the lack of concrete evidence regarding Marco Polo's visit to China. Some argue that Polo obtained information through contact with Persian traders. The answer provided is 'Through contact with Persian traders.' This answer is extracted from the following paragraph in the text: 'Marco Polo acquired much of his knowledge through contact with Persian traders since many of the places he named were in Persian.' This paragraph states that many of the places Polo observed in China had Persian names, serving as evidence of his visit. However, this evidence is not conclusive, and Polo could have acquired information about China from Persian traders. Figure 1 serves as a crucial piece of evidence supporting the main idea of our article. It highlights the necessity for NLP models to possess the capability to comprehend and analyze text for effectively answering complex and open-ended questions. In Figure 1, we observe a question and its answer from the SQuAD v2 dataset. The question explores how Marco Polo acquired knowledge about China, with the answer being through contact with Persian traders. This answer is derived from the following paragraph in the text: 'Marco Polo acquired much of his knowledge through contact with Persian traders since many of the places he named were in Persian.' This paragraph

Figure 1 Squad v2 Datasets Example

notes that the locations Polo observed in China often had Persian names, considered evidence of his visit. However, this evidence is inconclusive, leaving the possibility that Polo obtained information about China from Persian traders. This evidence emphasizes that NLP models must possess the ability to comprehend and analyze text for answering complex and open-ended questions effectively. Such questions demand the integration of information from different sections of the text, highlighting the importance of understanding and analyzing textual content. This evidence strongly supports the main idea of our article, which argues that NLP models must possess the capability to comprehend and analyze text for effectively answering complex and open-ended questions.

Model Architecture

BERT: BERT (Bidirectional Encoder Representations from Transformers) Fu *et al.* (2023) stands out as a pivotal achievement in natural language processing (NLP), having demonstrated considerable success. In the initial pre-training phase, the model acquires a broad understanding of language structure and context through unsupervised learning on extensive text corpora. Fine-tuning for specific tasks, such as question-answering (QA), further refines its capabilities, enabling accurate responses to posed questions within given text passages. Utilizing tokens like [CLS] (classification) and [SEP](separator) to structure input text in QA tasks, BERT's token embedding layers transform each word and token into numerical representations, enhancing overall understanding. The selfattention mechanism, inherent in its bidirectional attention design, amalgamates information from all words in the text to grasp the context of each word. Typically, outputs are derived from the [CLS] token and used for specific classification tasks, such as predicting the appropriate answer class for a given question. BERT's structured transformer network renders it adaptable to various NLP applications, showcasing particularly effective performance in question-answering tasks.

In this study, the research expands upon BERT's capabilities, focusing on its application in question-answering tasks. The model, pre-trained for the general QA task, is fine-tuned using the Squad v2 dataset. Similar to DistilBERT, BERT integrates both the question and context, drawing upon language representations obtained during pre-training to comprehend the information and generate precise answers to posed questions. The comprehensive examination of BERT's performance in the QA model, conducted with the Squad v2 dataset, is detailed within the Model Architecture section under the Methodology. This inclusion aims to elucidate BERT's distinctive role and capabilities in the context of this research.The steps and mathematical formula of the BERT model are shown in Figure 2.

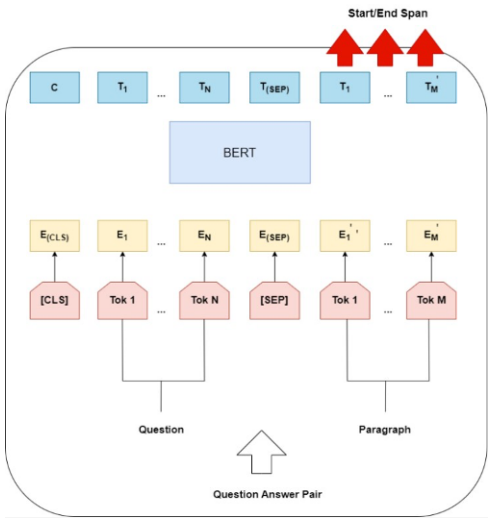


Figure 2 Use of BERT for question and answer

Figure 2 illustrates the functioning of the BERT QA model. The model is trained with a questionanswer pair consisting of a question designed to be posed by a human and a corresponding paragraph. The question is formulated in a way that seeks information, and the answer is based on the information present in the paragraph.

The model operates in two stages:

• In the first stage, the model converts the question and paragraph into a sequence of tokens. Tokens are the basic language elements that the model can comprehend. For instance, changing the question to "What is the capital of Turkey?" would result in tokens such as "Turkey," "capital," "is," and "?". • In the second stage, the model processes the tokens through a BERT model. BERT is a deep learning model with the ability to understand the context of a word in a sentence. The model is used to determine the meanings

of tokens and their relationships.

The text in the image defines different token types used by the model:

• Start/End Span tokens specify the start and end positions of the answer token in the paragraph. For example, for the question "What is the capital of Turkey?" the word "Ankara" could be an answer token, and the Start/End Span tokens would indicate its position in the paragraph.

• The CLS token indicates the question token. For instance, for the question "What is the capital of Turkey?" the CLS token would precede the word "Turkey."

• N tokens denote normal word tokens in the paragraph. For the question "What is the capital of Turkey?" the word "capital" would be an N token.

• M tokens denote stop-word tokens in the paragraph. For example, the "?" sign is an M token.

After processing the tokens, the model produces an answer token that can address the question. In this case, the answer token would be generated as "Ankara" based on the information present in the paragraph, specifying the position of "Ankara" with Start/End Span tokens.

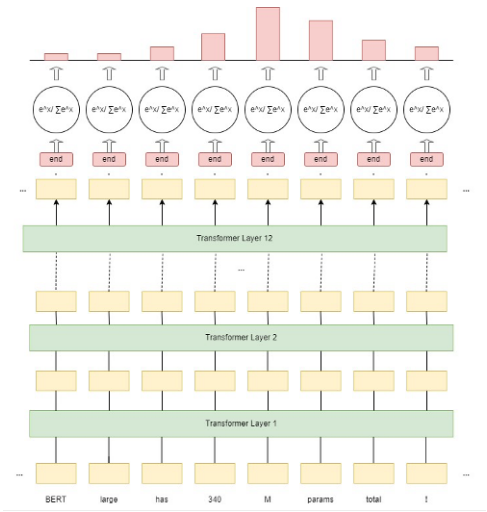


Figure 3 Layers passed in the Bert QA model

Figure 3 depicts the block diagram of the BERT QA model, revealing its three-tiered architecture. It illustrates the three layers that constitute the BERT model. The bottommost layer is the input layer, representing textual data. The middle layer serves as a transformative layer, employed to discern relationships among textual data. The top layer functions as an output layer, crucial for responding to queries.

• Input Layer : It is a vector representing textual data, encapsulating either a summary of the text document or the text question itself.

• Transformative Layer : This layer is dedicated to understanding the interconnections among textual data. It is intricately divided into twelve layers, interlinked through a pair of attention mechanisms.

• Attention Mechanisms : These mechanisms are utilized to ascertain the relationship of a specific point in the text with other textual segments. This aids the BERT model in comprehending the meaning of textual data.

• Output Layer : This layer consists of a vector used to respond to queries. This vector encapsulates either the answer to the question or a summary related to the answer.

This innovative depiction showcases the intricate design of the BERT QA model, emphasizing its capacity to unravel the complexities of textual relationships through attention mechanisms, ultimately providing insightful responses to posed questions.

Figure 4 depicts the internal structure of the Encoder and Decoder components of the BERT architecture.

The Encoder takes input from text or code and transforms it into a sequence of vectors. These vectors represent the meaning of the text and the position in the sentence. The Encoder consists of three main layers:

• Embedding Layer : This layer transforms each word into a vector.

• Multi-head Attention Layer : This layer learns relationships between words.

• Positional Encoding Layer : This layer represents the position of words in the sentence.

The Decoder takes vectors from the Encoder and produces the output. Similar to the Encoder, the Decoder comprises three main layers:

• Embedding Layer : This layer transforms each word into a vector.

• Multi-Head Attention Layer : This layer learns relationships between words.

• Positional Encoding Layer : This layer represents the position of words in the sentence.

Both Encoder and Decoder layers utilize a technique called attention mechanism. The attention mechanism is used to determine the relationship between a word and other words. This enables BERT to understand relationships between texts.

Fig.4 provides a detailed view of the internal structure of the Encoder and Decoder components. This information can help you better understand how the BERT architecture operates.

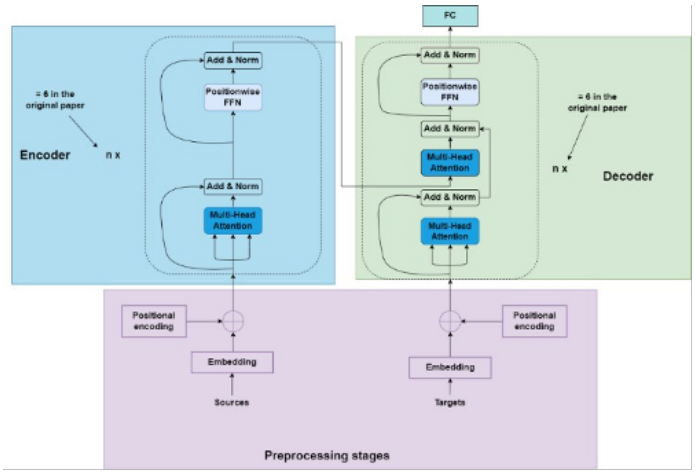


Figure 4 BERT QA Model Encoder-Decoder-Preprocessing Stages Relationship

Figure 4 illustrates the intricate relationship among the Encoder, Decoder, and Preprocessing Stages within the BERT architecture. This symbiotic connection is crucial for the effective functioning of BERT-based Question Answering (QA) models. Let's delve into the nuanced interplay between these components. The Preprocessing Stages serve as the initial gateway for input data. This phase involves tokenizing the text or code, converting words into vectors through the Embedding layer, and incorporating positional encoding to capture the contextual information of each word in the

sentence. The transformed input from the Preprocessing Stages then proceeds to the Encoder. The Embedding layer of the Encoder is pivotal in converting each tokenized word into a vector, providing a foundational representation of the input. The Multi-head attention layer follows suit, fostering an understanding of relationships between words. Simultaneously, the Positional encoding layer imparts valuable information regarding the spatial arrangement of words within the sentence. Moving on to the Decoder, it receives the enriched vectors from the Encoder and embarks on a similar journey. The Embedding layer refines the representations of words, while the Multi-head attention layer delves into the intricate web of relationships between words. The Positional encoding layer ensures that the positional information is retained throughout this process. The crux of this dynamic lies in the attention mechanism, a common thread woven into both the Encoder and Decoder. This mechanism empowers BERT to discern the significance of each word in relation to others, facilitating a holistic understanding of contextual dependencies.

In summary, the Preprocessing Stages lay the groundwork by transforming input data, the Encoder refines and enriches these representations, and the Decoder further refines them while decoding the final output. The cohesive interplay between these stages, guided by the attention mechanism, equips BERT-based QA models with the ability to grasp intricate relationships and nuances within the given text or code. This elucidation provides a concise yet comprehensive understanding of the intricate relationship between the Encoder, Decoder, and Preprocessing Stages as depicted in Figure 4.

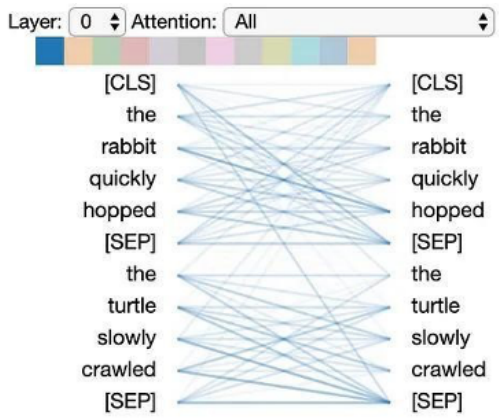


Figure 5 BERT Input Format and Textual

This textual example, [CLS] the rabbit quickly hopped [SEP] the turtle slowly crawled [SEP], can be elucidated as follows:

This instance represents the specific input format of the BERT model. The [CLS] token signifies the beginning and typically acts as a representative of the query. [SEP] tokens are used to separate two distinct text segments. The first [SEP] indicates the first text segment and the subsequent group of words. The second [SEP] denotes the second text segment and the following group of words.

In this example, the phrase "the rabbit quickly hopped" represents one text segment, while "the turtle slowly crawled" represents a second text segment. The differing relationships between them are based on the content of these two text segments. The occurrence of the rabbit swiftly hopping in the first segment implies one event, whereas the turtle slowly crawling in the second segment conveys a different event. BERT utilizes a learned attention mechanism to comprehend such relationships and grasp the context of text segments. This format serves as an example to showcase BERT's ability to understand relationships within text. When processing such inputs, the model can compare different text segments and decipher relationships. This explanation provides insight into the Figure 5, highlighting how BERT processes input data with distinct text segments, drawing attention to its learned attention mechanism for understanding textual relationships.

DistilBERT: DistilBERT is a distilled version of the BERT (Bidirectional Encoder Representations from Transformers) model, designed for natural language processing tasks such as Question Answering (QA). This model has been pre-trained to adapt to a customized QA task. DistilBERT is often characterized by smaller dimensions and parameters, designed to be smaller and faster than the original BERT model. However, this design choice may result in a slight loss of information compared to the original BERT model. Nevertheless, it offers advantages, especially in resource-constrained environments. Furthermore, it preserves the general learning capabilities of the original BERT model (Benedetto 2023).

In the QA task, the DistilBERT model focuses on providing answers to specific questions over a text paragraph or document. The model combines the question and context, utilizes the language representations it learned during pre-training to comprehend this information, and then generates an appropriate answer to the question. A pre-trained DistilBERT model, having general language understanding capabilities, can be fine-tuned for a specific QA task to be customized for better performance on a particular topic or dataset.

In this study, we trained DistilBERT with the Squad v2 dataset and describe its performance on the QA model. The introduction of DistilBERT within the Model Architecture section under the Methodology aims to provide a comprehensive understanding of its role and capabilities in the context of this research.

Figure 6's diagram elucidates the operational dynamics of the DistilBERT QA Model. The model processes inputs in two stages. In the first stage, inputs undergo tokenization and pass through an embedding layer, constructing a vector representation of the inputs. In the second stage, leveraging the vector representation of inputs, the model executes a question-answering task employing a series of transformer layers and a prediction layer.

• Inputs and Outputs : The model's inputs consist of a question and a context text. The question represents what we seek to know the answer to, while the context text embodies the text containing the answer to the question. The model's output is the answer to the question.

• Embedding Layer : This layer tokenizes inputs and generates a vector representation for each token, culminating in an overall vector representation of the inputs.

• Transformer Layers : Utilizing the vector representations of inputs, the model performs a question-answering task through a series of transformer layers. Each layer employs an attention mechanism, aiding in discerning relationships between vector representations of inputs.

• Attention Mechanism : This mechanism contributes to determining relationships between vector representations of inputs, enhancing the model's understanding of the contextual nuances.

• Prediction Layer : This layer predicts the answer to the question, consolidating the model's comprehension and providing a valuable output.

This innovative depiction underscores the intricate process of the DistilBert QA model, showcasing its proficiency in discerning relationships and delivering accurate responses to posed questions through the adept utilization of attention mechanisms and

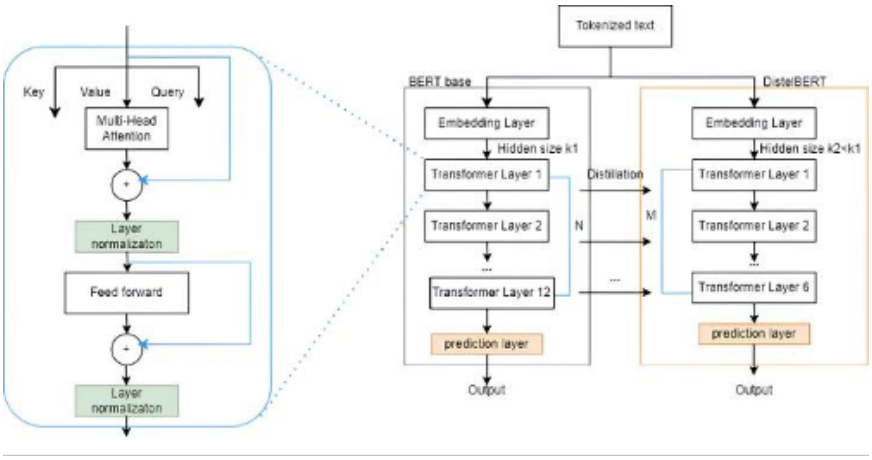


Figure 6 Example of how DistilBert works

transformer layers.

Finally, distinctions between the DistilBert QA model and the BERT base model can be outlined. Notably, the DistilBert model is smaller than the BERT base model. This implies that DistilBert has fewer parameters, resulting in a faster and more efficient operation. The DistilBert QA model proves to be a potent and efficient model for performing question-answering tasks. Its compact size, coupled with high performance, renders it suitable for a diverse range of applications.

RoBERTa: RoBERTa is a language model derived from BERT, building upon the masked language modeling (MLM) task at the core of BERT. However, RoBERTa exhibits significant differences from the original BERT model. Dynamic masking strategies are employed during the model's training, rendering the masking operations applied to input sequences more effective. This enhances the model's overall language understanding capability (Yasunaga *et al.* 2021).

A notable feature in RoBERTa's model architecture is the absence of the "Next Sentence Prediction" task. Instead, the model focuses on more specific tasks aimed at understanding relationships between sentences. This contributes to a better understanding of the context between sentences.

During training, RoBERTa is nourished with large datasets such as BooksCorpus, CC-News, OpenWebText, enriching the model's language learning abilities. Consequently, the model acquires more robust representation capabilities over general language knowledge.

In the process of model integration and fine-tuning, RoBERTa is adapted to be fine-tuned for a specific task. This customization enables the model to exhibit improved performance in the target task using pretrained language representations. The foundational architecture of RoBERTa, with its adaptability to specific tasks and overall language understanding capabilities, makes it an effective tool for various natural language processing tasks.

Figure 7 illustrates the performance of the RoBERTa QA model in an antonym test. In this test, the model is presented with both an original question and its antonym counterpart—a question that is opposite in meaning to the original. For instance, if the original question is 'What is the capital of Turkey?' the antonym question might be 'What is not the capital of Turkey?'

The horizontal axis in the Figure 7 represents the accuracy rate of the original question, while the vertical axis represents the accuracy rate of the antonym question. As evident from the graph, the RoBERTa QA model exhibits a lower accuracy rate in the antonym test compared to the original question. This indicates that the model encounters more challenges in comprehending antonym questions and providing accurate responses.

Several factors may contribute to this observation. Firstly, the model might have learned that antonym questions are inherently more difficult than the original questions. Secondly, the model could face difficulties in grasping the meaning of antonym questions. Lastly, the model may not have effectively learned the diverse strategies required to answer antonym questions.

These findings underscore the necessity for enhancing the resilience of the RoBERTa QA model specifically against antonym questions. This improvement could be achieved through the model gaining a better understanding of the meaning of antonym questions, developing different strategies, or undergoing specialized training for handling antonym questions. Addressing these aspects would contribute to the model's overall robustness in handling questions related to antonyms.

ALBERT: ALBERT, often referred to as "A Lite BERT," is specifically designed for natural language processing (NLP) tasks. It aims to lighten the BERT model to make it more scalable. While preserving the learning capabilities of BERT, ALBERT achieves the training of larger models with fewer parameters by factorizing the parameters in the embedding layer. This factorization allows the model to learn more efficiently. ALBERT is particularly developed to operate effectively and efficiently in resource-constrained environments. Additionally, it includes variants with different model sizes (small, base, large, xlarge), providing users the flexibility to choose the model size based on their needs (An *et al.* 2023).

In essence, the ALBERT model endeavors to deliver more effective performance in widely used natural language processing tasks by maintaining the advantages of BERT while presenting a lighter and more scalable architecture. ALBERT can be a suitable option, especially for applications that need to operate in resource-limited devices or environments.

In this study, ALBERT is trained with the Squad v2 dataset, and its performance on the QA model is elaborated upon. The introduction of ALBERT within the Model Architecture section under the Methodology aims to provide a comprehensive understanding of its role and capabilities within the scope of this research.

Flow diagram in Figure 8 illustrates the training process of the ALBERT QA model. The model undergoes a two-stage training process:

- Pre-training : During the pre-training stage, the model is

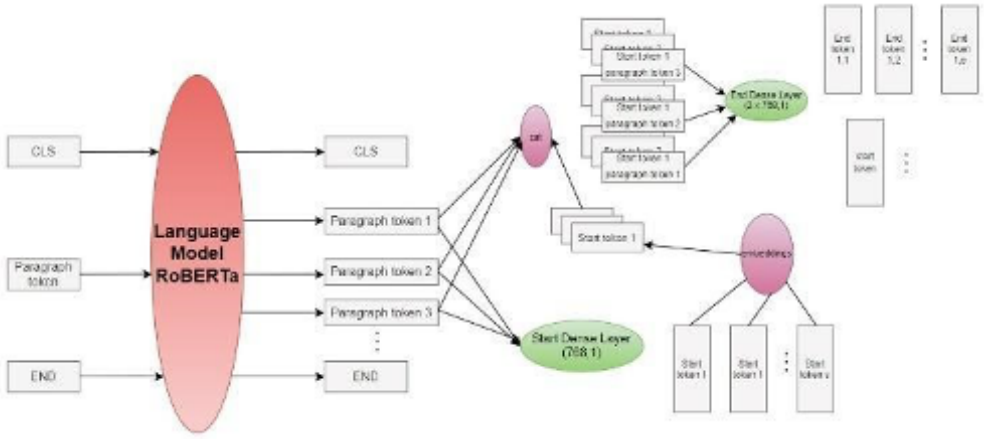


Figure 7 RoBERTa Model

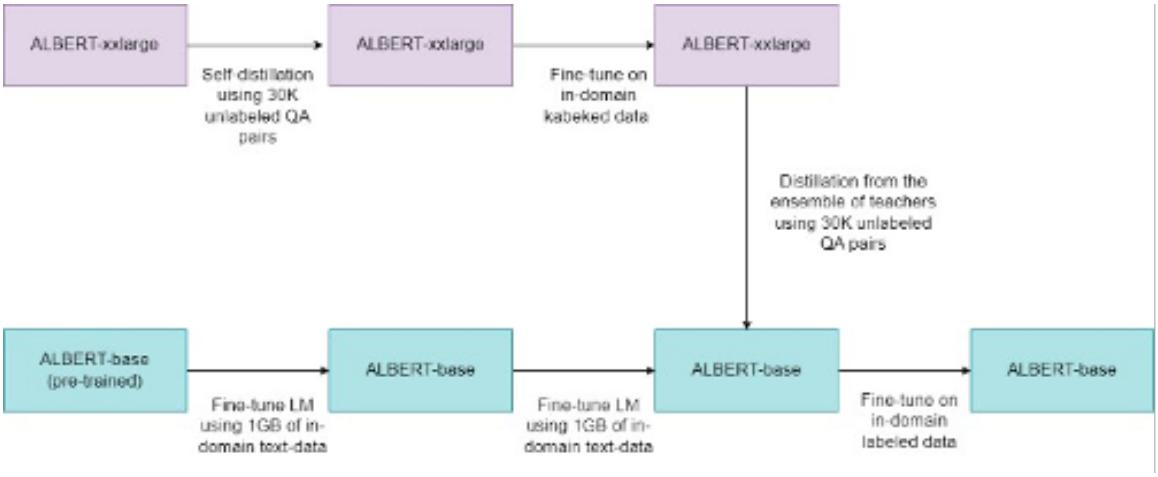


Figure 8 ALBERT QA Model Example

trained on a dataset comprising unlabeled text and question pairs. This dataset should encapsulate a representation of realworld questions and answers. By training on this dataset, the model learns to understand the relationships between text and questions (Tripathy *et al.* 2021).

• Fine-tuning : In the fine-tuning stage, the model is trained on a labeled dataset. This dataset includes question-answer pairs alongside accuracy labels. Training on this dataset allows the model to learn how to process text to correctly answer questions (Tripathy *et al.* 2022).

The pre-training stage enables the model to grasp its fundamental features. Here, the model learns to understand the relationships between text and questions, facilitating its comprehension to accurately answer questions.

In Figure 8, the pre-training stage is depicted as follows:

• ALBERT-xxlarge : The largest version of the model with 137B parameters.

• Self-distillation : The process of the model training on itself, employed to enhance its performance.

• 30K unlabeled QA pairs : The number of unlabeled question-answer pairs used for the pretraining of the model.

Self-distillation is a crucial process during which one version of the model is used to train another, contributing to the model's improved representation.

The subsequent fine-tuning stage allows the model to enhance its performance in a specific domain. In this stage, the model is trained on a labeled dataset that includes question-answer pairs and accuracy labels.

In Figure 8, the fine-tuning stage is depicted as follows:

• ALBERT-base : A smaller version of the model with 117B parameters.

• Fine-tune on in-domain labeled data : The process of the model training on itself, employed to enhance its performance.

• 30K unlabeled QA pairs : The process of training the model on a labeled dataset to ensure accurate answers to questions within a specific domain. The size of the labeled dataset used for the model's fine-tuning, significantly impacting its performance. A larger dataset contributes to the model's improved performance.

Training Process

BERT, ALBERT, and DistilBERT models were trained using the Squad v2 dataset. The training process involved fine-tuning pre-trained models to adapt their general language understanding capabilities to a specific QA task. Training data consisted of text paragraphs from the Squad v2 dataset along with various question-answer pairs directed towards these paragraphs. This process aimed to enhance the models' performance in specific topics or contexts.

Performance Metrics

The evaluation of BERT, ALBERT, and DistilBERT models was conducted using QA tasks on the Squad v2 dataset, and the assessment criteria were refined to encompass novel performance metrics. In addition to the traditional measures of accuracy, precision, recall, and F1 score, the evaluation now includes more specific metrics tailored to the nature of question-answering tasks.

The revised metrics comprise the following key elements:

• HasAns_exact : The count of responses that precisely provide an exact answer to the posed question.

• NoAns_exact : The count of responses that do not furnish an exact answer to the question.

• HasAns_partial : The count of responses that offer a partial answer to the question.

• NoAns_partial : The count of responses that fail to provide a partial answer to the question.

• HasAns_not_found : The count of responses that do not present any answer to the question.,

• Exact_match : The proportion of responses that provide an exact answer to the question.

Additionally, novel metrics are introduced to offer a more nuanced evaluation:

• F1_score : The proportion of responses that deliver both a correct and complete answer to the question.

• Best_f1_tresh : The threshold value at which the F1_score is maximized.

• Best_f1 : The highest value of the F1_score.

• Best_exact : The highest value of the Exact_match.

• Best_exact_tresh : The threshold value at which the Exact_match is maximized.

These refined metrics provide a comprehensive and nuanced assessment of the models' performance on the Squad v2 dataset, offering insights into their abilities to produce exact and partial answers, as well as highlighting their performance at different threshold values. This enhanced evaluation aims to better capture the distinctive features of each model and contribute to a more nuanced understanding of their effectiveness in addressing the challenges posed by the Squad v2 dataset.

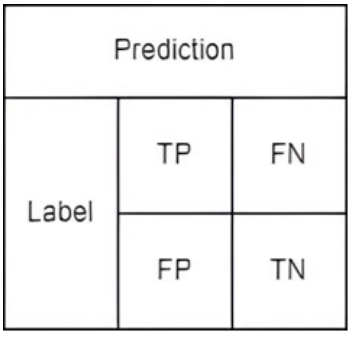


Figure 9 Confusion Matrix Explained

Accuracy Value: The accuracy value is a crucial performance metric that measures the percentage of correct predictions among the total predictions made by a model. It is mathematically expressed as:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

• TP (True Positive) : The number of correct positive predictions.

• TN (True Negative) : The number of correct negative predictions.

• FP (False Positive) : The number of incorrect positive predictions.

• FN (False Negative) : The number of incorrect negative predictions.

Accuracy is employed to assess the overall performance of a model. However, it can be misleading in the presence of imbalanced class distributions. This metric is crucial in evaluating how much of the model's predictions are correct. Especially when dealing with imbalanced datasets, accuracy should be considered alongside other performance metrics. What is TP, TN, FP, FN is explained in Figure 9.

F1 Score: The F1 Score is a significant performance metric that strikes a balance between precision and recall, providing a comprehensive evaluation of a model's ability to make accurate positive predictions while minimizing false positives and false negatives. Mathematically, it is expressed as:

$$F1\,\text{Score} = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{2}$$

Here:

• Precision: Measures the accuracy of positive predictions, calculated as

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3}$$

• Recall: Gauges the model's ability to capture all positive instances, calculated as

$$\text{Recall} = \frac{TP}{TP + FN} \tag{4}$$

RESULTS

During the evaluation of QA models on the SQuAD v2 dataset, each trained for 3 epochs, ALBERT emerged as the top performer, achieving an impressive 86.85% exact match and an 89.91% F1 score. Notably, ALBERT excelled in both answerable ('HasAns') and unanswerable ('NoAns') questions, demonstrating superior capabilities in providing accurate and comprehensive responses. BERT closely followed, demonstrating strong performance with a 65.96% exact match and a 70.12% F1 score. Its proficiency is particularly evident in answerable questions, where it achieved a remarkable 76.13% F1 score. RoBERTa secured robust results, yielding a 79.87% exact match and an 82.91% F1 score. Its balanced performance across answerable and unanswerable questions underscores its reliability in diverse QA scenarios. DistilBERT, while exhibiting competitive results with a 64.89% exact match and a 68.18% F1 score, falls slightly behind the other models. Nevertheless, it provides valuable insights, especially in scenarios where computational resources are constrained. In summary, the comprehensive assessment showcases ALBERT as the standout performer, followed closely by BERT and RoBERTa, while DistilBERT, although slightly trailing, remains a viable option for resource-efficient applications. You can see the values we have discussed and explained here in Table 1 comparatively.

Table 1 Results of All Models

Model	BERT-medium	DistilBERT	RoBERTa	ALBERT
F1	70.1163	68.1776	82.9125	89.9148
exact	65.9564	64.8898	79.8703	86.8525
total	11873	6078	11873	11873
HasAns_exact	67.7969	69.7595	77.9352	84.4467
HasAns_f1	76.1287	76.6267	84.0284	90.5801
HasAns_total	5928	2910	5928	5928
NoAns_exact	64.1211	60.4167	81.7998	89.2515
NoAns_f1	64.1211	60.4167	81.7998	89.2515
NoAns_total	5945	3168	5945	5945
Best_exact	65.9648	64.8898	79.8703	87.4168
Best_exact_thresh	0	0	0.95	-3.0903
Best_f1	70.1247	68.1776	82.9125	90.3287
Best_f1_thresh	0	0	0.95	-3.0903

CONCLUSION

The performances of DistilBERT, BERT, ALBERT, and RoBERTa models on the SQuAD v2 dataset were evaluated after three epochs of training in this study. The 'exact' and 'f1' scores were analyzed to compare the models. DistilBERT was found to lag slightly behind other models, exhibiting 'exact' and 'f1' scores of 64.89% and 68.18%, respectively. Despite this, it emerged as a valuable option in scenarios where computational resources are limited. Higher 'exact' and 'f1' scores were observed for BERT, standing out at 65.96% and 70.12%, respectively. Particularly impressive was its performance in answerable questions, contributing to an overall increased F1 score. ALBERT drew attention with stellar performance, demonstrating 'exact' and 'f1' scores of 86.85% and 89.91%, respectively. It showcased significant superiority over other models in both answerable and unanswerable questions. RoBERTa exhibited a balanced performance, achieving 'exact' and 'f1' scores of 79.87% and 82.91%, respectively, with consistent results for both answerable and unanswerable questions. The limited number of epochs may have contributed to DistilBERT generally showing lower performance, while the ALBERT model, showcasing rapid learning ability with a small number of epochs, surpassed other models.

BERT's transformer architecture allows it to achieve state-of-the-art results by pre-training on a large corpus and fine-tuning on specific tasks. The model employs a bidirectional approach, considering the context from both directions, which sets it apart from traditional models like LSTM and GRU. The SQuAD (Stanford Question Answering Dataset) has been a benchmark for evaluating question-answering models. The release of SQuAD v2 introduced unanswerable questions, adding a new challenge for models to determine when no answer is possible.

However, other factors should also be considered when making rankings. Model performances should be evaluated, particularly based on specific usage scenarios. For instance, while DistilBERT might be a valuable option in scenarios where computational resources are limited, BERT's impressive performance in answerable questions might make it a preferable choice in such scenarios. This study highlights the importance of hyperparameter optimization in fine-tuning pretrained models. The results suggest that fine-tuning BERT with appropriate hyperparameters can lead to significant performance gains. This insight contributes to the broader understanding of how pre-trained models can be effectively adapted to specific tasks.

In conclusion, an important guide for the use of models in various QA scenarios is provided by this evaluation, considering their performances. The advantages and disadvantages of each model should be assessed based on specific use cases.

In this study, the performances of four different BERT models - RoBERTa, ALBERT, DistilBERT, and BERT - in the field of Question Answering (QA) were compared after being trained for 3 epochs. Evaluation criteria included "exact" and "f1" scores. The obtained results indicate significant differences among the models. Initially, it was deemed necessary to rank the models based on the evaluation scores.

The ranking, according to the comparison values, is as follows: ALBERT - RoBERTa - BERT - DistilBERT. ALBERT, especially by demonstrating faster learning capability with fewer epochs, exhibits superior performance compared to other models. This advantage of ALBERT stems from the limited duration of the training. If the epoch numbers of other models are increased, these models might approach or surpass the accuracy rate achieved by ALBERT.

This emphasizes the importance of strategic decisions related to the training duration of models.

However, other factors should also be considered when making rankings. Model performances should be evaluated, particularly based on specific usage scenarios. For instance, while DistilBERT might be a valuable option in scenarios where computational resources are limited, BERT's impressive performance in answerable questions might make it a preferable choice in such scenarios. In conclusion, this evaluation provides a valuable resource for understanding the strengths and weaknesses of each model. It guides researchers in determining which model might be more effective in specific QA scenarios. Future studies could delve more deeply into how these models can be further enhanced with additional features and parameter adjustments and how they perform on different datasets.

Exploring ensemble approaches that combine BERT with other models could also yield interesting results. The implications of these findings are substantial for the development of more sophisticated AI-driven question-answering systems. By improving the model's ability to handle unanswerable questions, we enhance the reliability and user trust in AI systems. This has practical applications in customer service, virtual assistants, and information retrieval systems, where accurately identifying unanswerable questions can prevent misinformation and improve user experience.

Availability of data and material

Not applicable.

Conflicts of interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Ethical standard

The authors have no relevant financial or non-financial interests to disclose.

LITERATURE CITED

An, Q., B. Pan, Z. Liu, S. Du, and Y. Cui, 2023 Chinese named entity recognition in football based on albert-bilstm model. Applied Sciences **13**: 10814.

Benedetto, L., 2023 A quantitative study of nlp approaches to question difficulty estimation. In *International Conference on Artificial Intelligence in Education*, pp. 428–434, Springer Nature Switzerland.

Caballero, M., 2021 A brief survey of question answering systems. International Journal of Artificial Intelligence & Applications (IJAIA) **12**.

David, J., 2020 Comparing the representation learning of autoencoding transformer models in ad hoc information retrieval .

Devlin, J. *et al.*, 2018 Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 .

Fu, X., J. Du, H. T. Zheng, J. Li, C. Hou, *et al.*, 2023 Ss-bert: A semantic information selecting approach for open-domain question answering. Electronics **12**: 1692.

Ghanem, R., H. Erbay, and K. Bakour, 2023 Contents-based spam detection on social networks using roberta embedding and stacked blstm. SN Computer Science **4**: 380.

Gillioz, A., J. Casas, E. Mugellini, and O. Abou Khaled, 2020 Overview of the transformer-based models for nlp tasks. In *2020 15th Conference on Computer Science and Information Systems (FedCSIS)*, pp. 179–183, IEEE.

Greco, C. M., A. Tagarelli, and E. Zumpano, 2022 A comparison of transformer-based language models on nlp benchmarks. In *International Conference on Applications of Natural Language to Information Systems*, pp. 490–501, Springer International Publishing.

Kasai, J. *et al.*, 2024 Realtime qa: What's the answer right now? Advances in Neural Information Processing Systems **36**.

Kumar, A., T. Ranjan, and S. Raghav, 2023 Building conversational question answer machine and comparison of bert and its different variants. In *2023 Third International Conference on Secure Cyber Computing and Communication (ICSCCC)*, pp. 240–245, IEEE.

Kumari, V., S. Keshari, Y. Sharma, and L. Goel, 2022a Context-based question answering system with suggested questions. In *2022 12th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pp. 368–373, IEEE.

Kumari, V., Y. Sharma, and L. Goel, 2022b A comparative analysis of transformer-based models for document visual question answering. In *International Conference on Computational Intelligence and Data Engineering*, pp. 231–242, Springer Nature Singapore.

Lan, Z. *et al.*, 2019 Albert: A lite bert for self-supervised learning of language representations.

Liu, Y. *et al.*, 2019 Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 .

MacRae, C., 2022 *NOLEdge: Creating an Intelligent Search Tool for the Florida State University Computer Science Department Using Fine-Tuned Transformers and Data Augmentation*.

Malla, S. and P. J. A. Alphonse, 2021 Covid-19 outbreak: An ensemble pre-trained deep learning model for detecting informative tweets. Applied Soft Computing **107**: 107495.

Nassiri, K. and M. Akhloufi, 2023 Transformer models used for text-based question answering systems. Applied Intelligence **53**: 10602–10635.

Pereira, J., R. Fidalgo, R. Lotufo, and R. Nogueira, 2023 Visconde: Multi-document qa with gpt-3 and neural reranking. In *European Conference on Information Retrieval*, pp. 534–543, Springer Nature Switzerland.

Pirozelli, P., A. A. Brandão, S. M. Peres, and F. G. Cozman, 2022 To answer or not to answer? filtering questions for qa systems. In *Brazilian Conference on Intelligent Systems*, pp. 464–478, Springer International Publishing.

Rawat, A. and S. S. Samant, 2022 Comparative analysis of transformer based models for question answering. In *2022 2nd International Conference on Innovative Sustainable Computational Technologies (CISCT)*, pp. 1–6, IEEE.

Sabharwal, N. and A. Agrawal, 2021 *Hands-on Question Answering Systems with BERT: Applications in Neural Networks and Natural Language Processing*.

Schütz, M., A. Schindler, M. Siegel, and K. Nazemi, 2021 Automatic fake news detection with pretrained transformer models. In *Pattern Recognition. ICPR International Workshops and Challenges: Virtual Event, January 10-15, 2021, Proceedings, Part VII*, pp. 627–641, Springer International Publishing.

Sidorov, G., F. Balouchzahi, S. Butt, and A. Gelbukh, 2023 Regret and hope on transformers: An analysis of transformers on regret and hope speech detection datasets. Applied Sciences **13**: 3983.

Srivastava, V., S. Pilli, S. Bhat, N. Pedanekar, and S. Karande, 2021 What berts and gpts know about your brand? probing contextual language models for affect associations. In *Proceedings of Deep Learning Inside Out (DeeLIO): The 2nd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pp. 119–128.

Sundelin, C., 2023 Comparing different transformer models' performance for identifying toxic language online .

Tahsin Mayeesha, T., A. Md Sarwar, and R. M. Rahman, 2021 Deep learning based question answering system in bengali. Journal of Information and Telecommunication **5**: 145–178.

Tripathy, J. K., S. S. Chakkaravarthy, S. C. Satapathy, M. Sahoo, and V. Vaidehi, 2022 Albert-based fine-tuning model for cyberbullying analysis. Multimedia Systems **28**: 1941–1949.

Tripathy, J. K., S. C. Sethuraman, M. V. Cruz, A. Namburu, P. Mangalraj, *et al.*, 2021 Comprehensive analysis of embeddings and pre-training in nlp. Computer Science Review **42**: 100433.

Wang, N., R. R. Issa, and C. J. Anumba, 2022 Nlp-based query-answering system for information extraction from building information models. Journal of computing in civil engineering **36**: 04022004.

Yang, Z. *et al.*, 2019 Xlnet: Generalized autoregressive pretraining for language understanding. Advances in neural information processing systems **32**.

Yasunaga, M., H. Ren, A. Bosselut, P. Liang, and J. Leskovec, 2021 Qa-gnn: Reasoning with language models and knowledge graphs for question answering. arXiv preprint arXiv:2104.06378 .

Yu, M. and A. Sun, 2023 Dataset versus reality: Understanding model performance from the perspective of information need. Journal of the Association for Information Science and Technology **74**: 1293–1306.

How to cite this article: Ozkurt, C. Comparative Analysis of State-of-the-Art Q&A Models: BERT, RoBERTa, DistilBERT, and ALBERT on SQuAD v2 Dataset. *Chaos and Fractals*, 1(1), 19-30, 2024.

ADBA Chaos and Fractals
Cutting-Edge Scientific Solutions

e-ISSN: 3023-8951
RESEARCH ARTICLE
Vol.1 / No.1 / 2024 / pp.31-37
https://doi.org/10.69882/adba.chf.2024074

Analysis of a Resistive-Capacitive Shunted Josephson Junction with Topologically Nontrivial Barrier Coupled to a RLC Resonator

Jules Metsebo [*,1], **Boubakary Abdou** [β,2], **Dianorré Tokoue Ngatcha** [α,3], **Isidore Komofor Ngongiah** [θ,4], **Paul Didier Kamdem Kuate** [γ,5] **and Justin Roger Mboupda Pone** [σ,6]

[*]Department of Hydraulics and Water Management, National Advanced School of Engineering, University of Maroua, P.O Box 58 Maroua, Cameroon, [β]Department of Mechanical, Petroleum and Gas Engineering, National Advanced School of Mines and Petroleum Industries, University of Maroua, P.O. Box 46, Maroua, Cameroon, [α]Laboratory of Mechatronics, Energiatronics and Sustainable Mobility, Department of Automotive and Mechatronics Engineering, National Higher Polytechnic School of Douala, University of Douala, P.O. Box 24, 2701 Douala, Cameroon, [θ]Department of Physics, Faculty of Science, University of Bamenda, P.O. Box 39 Bamenda, Cameroon, [γ]Laboratory of Condensed Matter, Electronics and Signal Processing, Department of Physics, University of Dschang, P.O. Box 067, Dschang, Cameroon, [σ]Research Unit of Automation and Applied Computer (RU-AIA), Electrical Engineering Department of IUT-FV, University of Dschang, P.O. Box: 134, Bandjoun, Cameroon.

ABSTRACT This paper presents the resistive-capacitive shunted Josephson junction (RCSJJ) with a topologically nontrivial barrier (TNB) coupled to a linear RLC resonator. The rate equations describing RCSJJ with TNB coupled to the linear RLC resonator are established via Kirchhoff's current and voltage laws. The model exhibits four, two, or no equilibrium points depending on the external direct current (DC) source and the fractional parameter. The stability analysis of the equilibrium points with credit to the Routh-Hurwitz stability criterion reveals that the stability of equilibrium points depends on the DC source and the fractional parameter. Current-voltage characteristic reveals the presence of a birhythmicity zone which is sensitive to the fractional parameter m. As the fractional parameter increases, the coexistence of the resonant state is destroyed, which is followed simultaneously by the appearance of a new resonance state. Depending on initial conditions, birhythmic behaviour is characterized by the existence of a limit cycle. The projection of the phase space in the specific plane and the time evolution of charge is predicted in which the amplitude of attractors reported is sensitive to the parameter m. Lastly, with a defined fractional parameter, the amplitude of the branch locked to the resonator is greater than the unlocked branch.

KEYWORDS
Josephson junction
Topologically nontrivial barrier
RLC resonator
Birhythmic zone
Limit cycle
Resonant state

INTRODUCTION

In 1962, Brian David Josephson analysed the happenings at the junction between two closely spaced superconductors, separated by an insulating barrier. If the insulating barrier is thick, the electron pairs cannot get through, but if the layer is thin enough (approximately 10 nm) there is a probability for electron pairs to the tunnel (Fujii *et al.* 2009). This effect later became known as "Josephson tunnelling". Whatever the nature of the junction, its thickness has to be comparable to or smaller than the coherence length of the two superconductors (Owen and Scalapino 1997). Otherwise, the dynamics of the respective Cooper pairs are uncorrelated.

Manuscript received: 11 May 2024,
Revised: 27 June 2024,
Accepted: 27 July 2024.

[1]jmetsebo@gmail.com (**Corresponding author**)
[2]ababakrys@yahoo.com
[3] tokouengatcha1@gmail.com
[4]ngongiahisidore@gmail.com
[5]kamdempauldiddy@gmail.com
[6]mboupdapone00@gmail.com

The current that crosses the barrier is the Josephson current. The Josephson effect is based on the behaviour of a quantum parameter called phase. As a result of the fact that the barrier is thin enough, the phase of the electron wave function in one superconductor maintains a fixed relationship with the phase of the wave function in the other superconductor (Eck *et al.* 1965). This linking up of phases is called phase coherence which is the essence of the Josephson effects. Recently, the interest in Josephson physics has shifted towards junctions whose elements include topological materials Bao *et al.* (2015) which enable the specific design of the hybrid junction (Mourik *et al.* 2012).

In the literature, JJs with a TNB is a model that can display excitable mode, bistable, periodic and chaotic behaviours (Kingni *et al.* 2020). In essence, there is an interesting aspect of topological nontrivial barriers known as fractality that plays an important role in several physical phenomena such as the nuclear fusion that occurs in main progression stars like the sun and this is known as quantum tunnelling (Bee *et al.* 2008). Quantum tunnelling in JJ has been the subject of great interest because of its possible technological applications in phase qubit Nori (2008); Clarke and Wilhelm

(2008), superconducting electronic devices, ultrahigh sensitive detectors, parametric amplifiers, voltage standards and superconducting quantum interference devices for the detection of very weak magnetic fields (Kleiner *et al.* 2004; McDermott *et al.* 2004; Oladapo *et al.* 2018). In the literature, JJs based on low-critical temperature superconductors have a harmonic current phase relation (Belli *et al.* 2012). While JJs based on high critical-temperature superconductors Campi *et al.* (2021) have an unharmonic current-phase relation due to the anisotropic and multiband effects in superconductors.

The analytical and numerical analysis of JJs with unharmonic current-phase relation has been reported in the literature over the past year (Canturk and Askerzade 2013; Kingni *et al.* 2019). In the case of JJs based on topologically nontrivial barriers Veldhorst *et al.* (2012), the current-phase relation includes an additional fractional term related to the Majorana particles (Dominguez *et al.* 2012; Fu and Kane 2009). Majorana particles are particles identical to their antiparticles which are described by real value wave functions and attract considerable interest in quantum computations (Lima *et al.* 2021; Wendin 2017). Furthermore, topological supercurrents interaction and fluctuations in the multiterminal Josephson effect emerge from the quantum coherence of electron trajectories and are sensitive to changes in the external magnetic field or gate voltage (Xie and Levchenko 2019; Ramakrishnan *et al.* 2022). When superconductivity is induced at the boundary of the mesoscopic sample via the proximity effect, the universality of conductance fluctuations remains intact (Bao *et al.* 2015; Aleiner and Blanter 2002). The phenomenon of conductance fluctuations is an aspect of fractality which is based on the fractional quantum Hall effect and appears in electron systems in two-dimensional (2D) for a strong magnetic field (Wen 2006).

The contemporary presence of two frequencies for the same set of parameters, or birhythmicity, is encountered in some biochemical systems Decroly and Goldbeter (1982); Morita *et al.* (1989); Haberichter *et al.* (2001); Sosnovtseva *et al.* (2002); Abou-Jaoudé *et al.* (2011a), nonlinear electronic circuits Kadji *et al.* (2007); Zakharova *et al.* (2010); Yamapi *et al.* (2010); Ghosh *et al.* (2011); Yamapi *et al.* (2012); Yue *et al.* (2012), and extended distributed systems (Stich *et al.* 2002; Casagrande and Mikhailov 2005). The experimental observation of birhythmic systems, however, is less frequent (Hounsgaard *et al.* 1988; Geva-zatorsky *et al.* 2006; Ventura *et al.* 2007; González *et al.* 2008).

In this context, the superconducting circuit consisting of JJs coupled to a cavity Hadley *et al.* (1988); Filatrella *et al.* (1992); Ozyuzer *et al.* (2007), represents a preeminent example of a birhythmic system that is also interesting for applications. The coupling among the junctions is supposed to be provided by a resonant cavity Gross *et al.* (2013); Grib *et al.* (2006), thus when all the junctions are entrained it is essential to have a large current in the cavity such that the junctions can be entrained through the current in the resonator. The state with a large current coexists with a state at lower power, such that the two states are characterized by two different frequencies. This is the essential feature of birhythmicity, the coexistence of two attractors characterized by two different amplitudes and frequencies.

Birhythmicity is therefore a nonlocal phenomenon that cannot be investigated by linear analysis. The global stability of the two states at different frequencies of the current-voltage (IV) characteristic ascertained the birhythmic properties induced by the RLC circuit. There is evidence from simulation that at the same bias point, two frequencies appear, depending on the initial conditions. The first frequency is reached by increasing the bias current from zero on the Josephson supercurrent, while the second is obtained by decreasing the current from high values on the resistive McCumber branch, such that the selection of the frequency displayed is thus determined by the initial conditions (Yamapi and Filatrella 2014). The features of the IV depend upon other factors such as the number of JJs and the features of the resonator (Grib *et al.* 2006). Also, heating effects are believed to be relevant for coherent radiation Wang *et al.* (2010), as well as coupling through charge transfer through the Josephson channel (Ovchinnikov and Kresin 2013).

The JJ can be considered a birhythmic system because it can produce oscillations at two distinct periods (Barone and Paternó 1982). Birhythmicity is encountered in some biochemical Abou-Jaoudé *et al.* (2011b) and non-linear electronic systems (Yamapi *et al.* 2010; Ghosh *et al.* 2011; Yamapi *et al.* 2012; Yue *et al.* 2012). In JJ physics, it is encountered in arrays coupled through an external circuit that possesses resonances (Likharev 1986). In the literature, it is well known that the junctions are hysteretic, and hence can be considered birhythmic, which is capable of oscillating at different frequencies for the same set of parameters. In this regard, Pountougnigni *et al.* (2019) presented the effects of uncorrelated white noise, in a series of JJs coupled to a linear RLC resonator.

Kingni *et al.* (2020) analysed the aspect of JJ with TNB. Nana *et al.* (2018) investigated the dynamics of an RLC series circuit with a hysteretic iron-core inductor, and not limited to Yamapi and Filatrella (2014) who investigated the effect of noise on a JJ that is coupled to a linear RLC resonator. Based on this literature and to the best of our knowledge, there is no work in the literature on the analytical and numerical analysis of an RCSJJ with TNB coupled to a linear RLC resonator. This paper is aimed at investigating the analytical and numerical analyses of an RCSJJ with TNB coupled to a linear RLC resonator. This leads to the following specific objectives: To model and analyse an RCSJJ with TNB coupled to a linear RLC resonator and to characterise the dynamical behaviours exhibited by an RCSJJ with TNB coupled to a linear RLC resonator. This paper is divided into three sections. Section 1 deals with the introduction of the paper. Section 2 presents the rate equations and theoretical analyses of the RCSJJ with TNB coupled to a linear RLC resonator. Finally, Section 3 presents the conclusion.

THEORETICAL ANALYSIS OF THE RCSJJ WITH A TNB COUPLED TO A LINEAR RLC RESONATOR

The RCSJJ with TNB coupled to a linear RLC resonator is an electrical circuit depicted in Fig. 1.

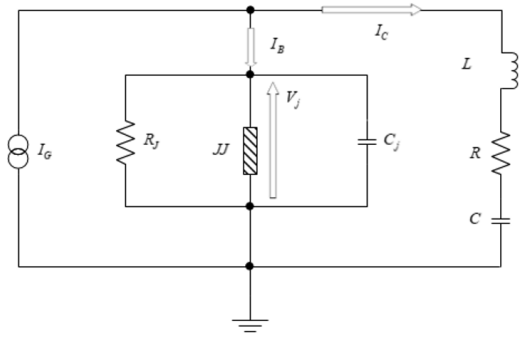


Figure 1 Equivalent electric circuit of the RCSJJ with TNB coupled to a linear RLC resonator.

Applying Kirchhoff's current and voltage laws to the schematic of Fig. 1, one has the following set of equations:

$$\begin{cases} I_G = I_{R_J} + I_J + I_C + I, \\ V_J = V_R + V_L + V_C, \\ V_J = \dfrac{\hbar}{2e}\dfrac{d\phi}{dt} \end{cases} \tag{1}$$

where $I_{R_J}, I_J = I_{J_C}\sin\phi + I_{UCPR}\sin\left(\frac{\phi}{2}\right)$ and I_C are the currents through the resistor across the junction, the JJ, and the capacitor across the junction respectively. I_{J_C} is the critical current of the junction, and $\phi = \phi_2 - \phi_1$ is the gauge-invariant phase difference across the two superconductors. Also, V_R, V_L, V_C, and I are the voltages across the resistor, inductor, capacitor, and the current on the RLC loop respectively and by substituting the various expressions of the currents and the expression of the voltages in the system (1), the following system (2) is obtained:

$$I_G = \frac{\hbar C_J}{2e}\frac{d^2\phi}{dt^2} + \frac{\hbar}{2eR_J}\frac{d\phi}{dt} + I_{J_C}\sin\phi + I_{UCPR}\sin\left(\frac{\phi}{2}\right) + \frac{d\tilde{q}}{dt}, \tag{2}$$

$$\frac{d^2\tilde{q}}{dt^2} + \frac{R}{L}\frac{d\tilde{q}}{dt} - \frac{\hbar}{2eL}\frac{d\phi}{dt} + \frac{1}{CL}\tilde{q} = 0. \tag{3}$$

Introducing the following dimensionless parameters,

$$\omega_o = \sqrt{\frac{2eI_{JC}}{\hbar C_J}}, \quad \tau = \omega_o t, \quad q = \omega_o\tilde{q}/I_{JC},$$

$$\alpha = \frac{1}{R_J}\sqrt{\frac{\hbar}{2eI_{JC}C_J}}, \quad \beta_L = \frac{2eI_{JC}L}{\hbar}, \quad \Omega = \frac{1}{\omega_o\sqrt{LC}}, \quad \sigma = \frac{R}{L\omega_o},$$

$$m = \frac{I_{UCPR}}{I_{JC}}, \quad \gamma_G = \frac{I_G}{I_{JC}},$$

the normalized form of the system (2) is as follows:

$$\begin{cases} \dfrac{d^2\phi}{d\tau^2} + \alpha\dfrac{d\phi}{d\tau} + \sin\phi + m\sin\left(\dfrac{\phi}{2}\right) + \dfrac{dq}{d\tau} = \gamma_G, \\ \dfrac{d^2q}{d\tau^2} + \sigma\dfrac{dq}{d\tau} - \dfrac{1}{\beta_L}\dfrac{d\phi}{d\tau} + \Omega^2 q = 0 \end{cases} \tag{4}$$

It is interesting to note that the parameters; γ_G is the applied current, Ω is the resonant frequency, and β_L is the perturbed frequency of the junction, and the fractional parameter $m(m \geq 0)$ represents the contribution of 2π–periodic Josephson current for $m = 0$ or 2π–periodic Josephson current and 4π–periodic Josephson current for $0 < m \leq 1$. The current phase relation of the Josephson current cannot take into account only periodic the 4π–Josephson current. Therefore, it is a particular case of the current-phase relation used in [56]. Letting $v = \frac{d\phi}{d\tau}$ and $i = \frac{dq}{d\tau}$ system (3) is transformed into a system of first-order ordinary differential equations given by:

$$\frac{dv}{d\tau} = \gamma_G - \alpha v - i - \sin\phi - m\sin\left(\frac{\phi}{2}\right), \tag{5}$$

$$\frac{di}{d\tau} = \frac{1}{\beta_L}v - \sigma i - \Omega^2 q, \tag{6}$$

$$\frac{d\phi}{d\tau} = v, \tag{7}$$

$$\frac{dq}{d\tau} = i. \tag{8}$$

System (4) is the mathematical transformation of the schematic of Fig. 1 representing the RCSJJ with TNB coupled to a linear RLC resonator.

Stability analysis of the RCSJJ with TNB couple to a linear RLC resonator

The stability of the system is done by characterising the equilibrium points of the system. This is done by solving the system (4) for; $\frac{d\phi}{d\tau} = 0$, $\frac{dv}{d\tau} = 0$, $\frac{dq}{d\tau} = 0$, and $\frac{di}{d\tau} = 0$. By this resolution of the equilibrium points $E_i(v^*, i^*, \phi^*, q^*)$ system (4) is characterised by different values of the fractional parameters m which elaborate on the various states in which the system exists. When the fractional parameter $m = 0$, system (4) reduces to:

$$v^* = 0; \quad i^* = 0; \quad q^* = 0, \tag{9}$$

$$\gamma_G - \sin\phi^* = 0, \tag{10}$$

as shown in [57], with two solutions $E_1(0, 0, \arcsin(i), 0)$ and $E_2(0, 0, \pi - \arcsin(i), 0)$ for $\gamma_G \leq 1$, and no solution for $\gamma_G > 1$. The associated characteristic equation is given by:

$$\lambda^4 + b_3\lambda^3 + b_2\lambda^2 + b_1\lambda + b_0 = 0, \tag{11}$$

where

$$b_3 = \sigma + \alpha; \quad b_2 = \frac{\Omega^2\beta_L + \alpha\beta_L + \beta_L\cos(\phi^*) + 1}{\beta_L};$$

$$\mathrm{b}_1 = \alpha\Omega^2 + \sigma\cos(\phi^*); \quad b_0 = \Omega^2\cos(\phi^*).$$

For a system to be stable, it is necessary and sufficient that each turn of the first column of the Routh array must be positive. If this condition is not met, the system is unstable and the numbers of sign changes in the first column correspond to the number of roots of the characteristic equation in the right half of the S-plane. The first column of the Routh array for system (6) is given by the coefficients of the d_i given by $d_1 = b_3$; $d_2 = \frac{b_3b_2 - b_1}{b_3}$; $d_3 = b_1 + \frac{b_0b_3}{d_2}$; $d_4 = b_0$. After analysis, it is found that the equilibrium point E_1 is unconditionally stable while the equilibrium point E_2 is unstable.

When the fractional parameter is $0 < m \leq 1$, the resolution of the system (4) is given by:

$$v^* = 0; \quad i^* = 0; \quad q^* = 0, \tag{12}$$

$$\gamma_G - \sin\phi^* - m\sin\left(\frac{\phi^*}{2}\right) = 0. \tag{13}$$

System (7) is a nonlinear system in ϕ^*, which the solution can only be approximated via a numerical method say the Newton-Raphson method. By this, the system depicts no roots, two or four roots as a function of the value of parameters ($\gamma_G = i_{dc}, m$) as shown in Fig. 2.

The characteristic equation of system (4) evaluated at the equilibrium points $E_i(0, 0, \phi^*, 0)$ is given by Eqn. (6) with the following:

$$b_3 = \sigma + \alpha;$$

$$b_2 = \frac{1}{2}\left(\frac{m\beta_L\cos\left(\frac{\phi^*}{2}\right) + 2\Omega^2\beta_L + 2\sigma\alpha\beta_L + 2\beta_L\cos(\phi^*) + 2}{\beta_L}\right);$$

$$b_1 = \alpha\Omega^2 + \sigma\cos(\phi^*) + \frac{1}{2}\sigma m\cos\left(\frac{\phi^*}{2}\right);$$

$$b_0 = \frac{1}{2}\Omega^2\left(m\cos\left(\frac{\phi^*}{2}\right) + 2\cos(\phi^*)\right).$$

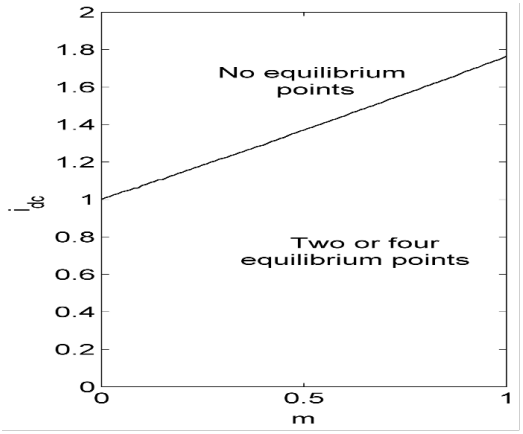


Figure 2 Repartition of equilibrium points of system (4) in the parameter space spanned by m and $\gamma_G = i_{dc}$.

Given credit to the Routh-Hurwitz stability criterion, the real parts of all the roots λ of Eqn. (6) are negative if and only if: $d_1 = b_3$; $d_2 = \frac{b_3 b_2 - b_1}{b_3}$; $d_3 = b_1 + \frac{b_0 b_3}{d_2}$; $d_4 = b_0$ are all greater than zero, or else the system is unstable.

Numerical analysis of the RCSJJ with TNB couple to a linear RLC resonator

The influence of the nontrivial barrier will be investigated in this section numerically. Figure 3 presents the voltage-intensity curves for different values of the parameter m.

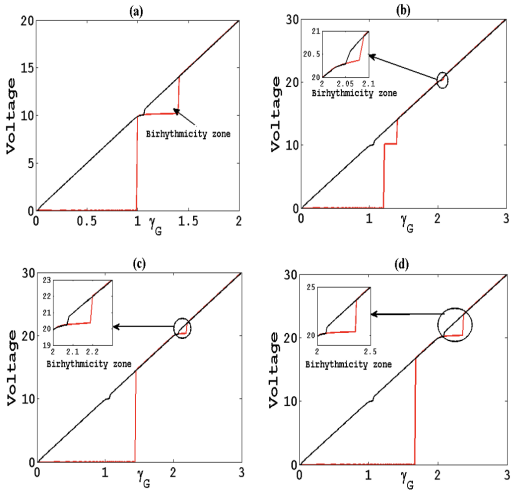


Figure 3 Voltage-intensity curves for different values of m: m_1, m_2, m_3, m_4. The red solid line represents the increasing current bias and the black solid line represents the decreasing current bias. The inset shows the birhythmicity zone for the RCSJJ with the TNB (m_5). The parameters of simulations are: α, β, γ, δ.

In Figure 3 (a) for $m = m_1$, the resonant step locked to the cavity is more detailed in the range $1.05 < \gamma_G < 1.40$ where the system stays on one or the other frequency, depending on the initial conditions (that are controlled by the bias sweep) [58]. The relevant feature is that the system remains birhythmic for any value of m as shown in Fig. 3 (b) to (d). Another feature inferred by the fractional parameter is the coexistence of two resonant state intervals for m bounded between m_2 and m_3. Therefore one realizes that the first resonant state interval $\gamma_G = \gamma_{G1}$ is still on the diagram and is slightly destroyed as m increases (see Fig. 3 (b)). The destruction of the former is followed simultaneously by the appearance of the new resonant state interval. The total disappearance of the first resonant state interval is evident when m is set beyond the value m_4 (see Fig. 3 (c) and Fig. 3 (d)). The latter is enlarged with the increase of m. The hysteresis on the new resonant state interval confers to the system the birhythmic properties. Figure 4 shows the projection of phase space in the (x, y) plane and the time evolution of the charge resonator for different values of parameter m. The birhythmic behaviour is characterized by the existence of two limit cycles depending on initial conditions. Each attractor can be identified by its amplitude and frequency.

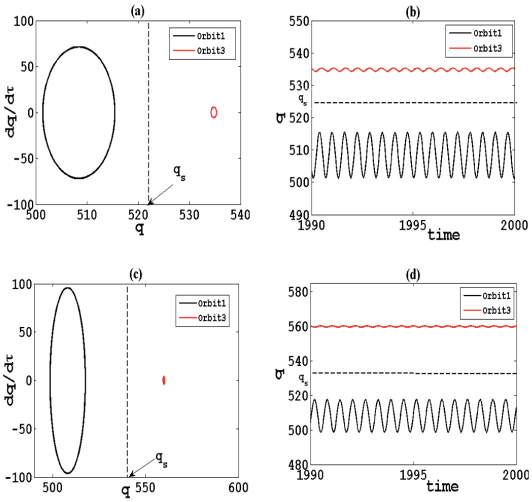


Figure 4 (a) and (b) Projection of phase space in $q - \frac{dq}{d\tau}$ plane and time evolution respectively for $m = 0.6$ and $\gamma_G = 2.15$; (c) and (d) projection of phase space in $q - \frac{dq}{d\tau}$ plane and time evolution respectively for $m = 0.9$ and $\gamma_G = 2.25$. The parameters of the simulation are $\alpha = 0.1$, $\beta_L = 0.01$, $\Omega = 2.0$, $Q = 200.0$.

Figure 4 is associated with the attractor with large excursion in charge oscillations with frequency Ω_1 and to the other attractor the frequency Ω_3 which is stable. One can estimate the existence of an unstable orbit with the frequency Ω_2 between the two which is indicated on the diagram by the guessed position q_s of the separatrix. Regarding the curves on the diagram, the role that can be assigned to the parameter m is described as follows: the orbit with the frequency Ω_1 sees its amplitude increasing with the increase of the parameter m while the amplitude of the orbit with frequency Ω_2 tends to smaller values. Another way of analyzing is to set the variation of the amplitude of voltage and the amplitude of charge when the bias is swept. We define the amplitude as the large excursion of the phase derivative $\frac{d\phi}{d\tau}$ (which is proportional to the JJ voltage) and the charge q (which is proportional to the capacitor voltage). Therefore, we set:

$$A = \max_{\tau} \frac{d\phi}{d\tau} - \min_{\tau} \frac{d\phi}{d\tau}, \tag{14}$$

$$B = \max_{\tau} q - \min_{\tau} q. \tag{15}$$

The amplitude diagram versus the bias current γ_G for many values of the parameter m is shown in Figs. 5 and 6.

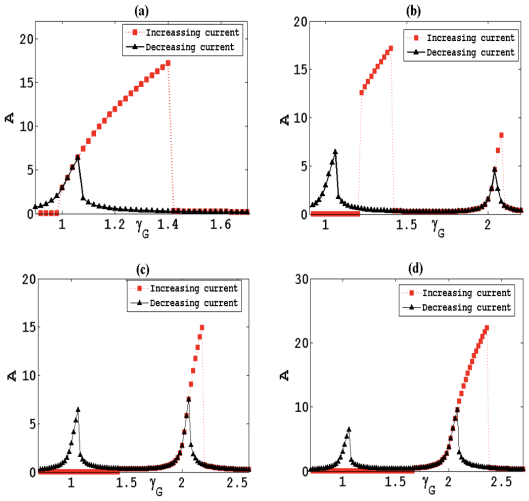


Figure 5 Amplitude A as a function of the bias current γ_G for many values of the parameter m: (a) $m = 0$, (b) $m = 0.3$, (c) $m = 0.6$, (d) $m = 0.9$. Squares refer to increasing the bias current and the triangles to decreasing the bias current which is equivalent to the voltage oscillations across the JJ. The parameters of simulations are the same as in Fig. 2.

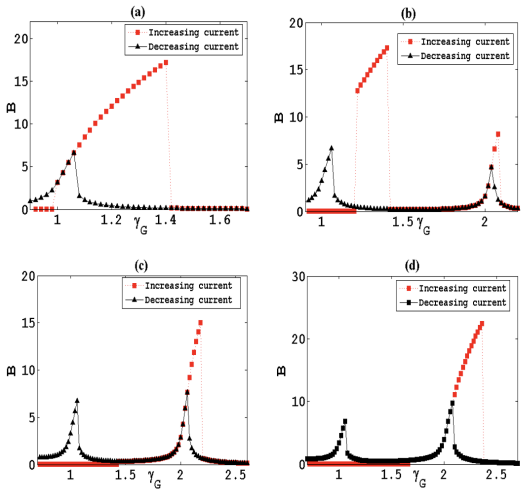


Figure 6 Amplitude B as a function of the bias current γ_G for many values of the parameter m: (a) $m = 0$, (b) $m = 0.3$, (c) $m = 0.6$, (d) $m = 0.9$. Squares refer to increasing the bias current and triangles refer to decreasing the bias current which is equivalent to the voltage of the capacitor. The parameters of simulations are the same as in Fig. 2.

CONCLUSION

This paper was devoted to the analytical and numerical analyses of an RCSJJ with a TNB coupled to a linear RLC resonator. The modelling and numerical investigation of the RCSJJ with TNB coupled to a linear RLC resonator were studied. The rate equations describing RCSJJ with TNB coupled to the RLC resonator are established by using Kirchhoff's current and voltage laws. The system was characterized by four, two, or no equilibrium points depending on the external direct current (DC) source and the fractional parameter.

The stability analysis of the equilibrium points with credit to the Routh-Hurwitz stability criterion reveals that the stability of equilibrium points depends on the system's parameters. Current-voltage characteristic reveals the presence of the birhythmicity zone which is sensitive to the fractional parameter m. As the fractional parameter increases, the coexistence of the resonant state is destroyed, which is followed simultaneously by the appearance of a new resonance state. Depending on initial conditions, birhythmic behaviour is characterized by the existence of period-1-attractors. The projection of the phase space in the specific plane and the time evolution of charge is predicted in which the amplitude of attractors reported are sensitive to the fractional parameter.

Availability of data and material

Not applicable.

Conflicts of interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Ethical standard

The authors have no relevant financial or non-financial interests to disclose.

LITERATURE CITED

Abou-Jaoudé, W., M. Chaves, and J.-L. Gouzé, 2011a A theoretical exploration of birhythmicity in the p53-mdm2 network. PLoS One **6**: e17075.

Abou-Jaoudé, W., M. Chaves, and J.-L. Gouzé, 2011b A theoretical exploration of birhythmicity in the p53-mdm2 network. PLoS One **6**: e17075.

Aleiner, I. L. and Y. M. Blanter, 2002 Inelastic scattering time for conductance fluctuations. Phys. Rev. B **65**: 1–10.

Bao, R., L. Huang, Y. Lai, and C. Grebogi, 2015 Conductance fluctuations in chaotic bilayer graphene quantum dots. Phys. Rev. E **012918**: 1–8.

Barone, A. and G. Paternó, 1982 *Physics and applications of the Josephson effect*. Wiley and Sons, New York.

Bee, G., K. Ballentine, and M. Thomsen, 2008 Realistic problems involving thermal conductivity. Am. J. Phys. **76**: 970–974.

Belli, F., T. Novoa, J. Contreras-García, and I. Errea, 2012 Strong correlation between electronic bonding network and critical temperature in hydrogen-based superconductors. Nat. Commun. **12**: 1–11.

Campi, D., S. Kumari, and N. Marzari, 2021 Prediction of phonon-mediated superconductivity with high critical temperature in the two-dimensional topological semimetal w2n3. Nano Lett. **21**: 3435–3442.

Canturk, M. and I. N. Askerzade, 2013 Chaotic dynamics of externally shunted josephson junction with unharmonic cpr. J. Supercond. Nov. Magn. **26**: 839–843.

Casagrande, V. and A. S. Mikhailov, 2005 Birhythmicity, synchronization, and turbulence in an oscillatory system with nonlocal inertial coupling. Phys. D Nonlinear Phenom. **205**: 154–169.

Clarke, J. and F. K. Wilhelm, 2008 Superconducting quantum bits. Nature **453**: 1031–1042.

Decroly, O. and A. Goldbeter, 1982 Birhythmicity, chaos, and other patterns of temporal self-organization in a multiply regulated biochemical system. Proc. Natl. Acad. Sci. USA **79**: 6917–6921.

Dominguez, F., F. Hassler, and G. Platero, 2012 Dynamical detection of majorana fermions in current-biased nanowires. Phys. Rev. B **86**: 1405031–5.

Eck, R., D. Scalapino, and B. Taylor, 1965 Low-temperature physics ix. In *9th International Conference*, edited by J. G. e. a. Daunt, p. 415, Plenum Press, Inc.

Filatrella, G., G. Rotoli, N. Gro/nbechJensen, R. D. Parmentier, and N. F. Pedersen, 1992 Model studies of long josephson junction arrays coupled to a high-q resonator. Journal Appl. Phys. **72**: 3179–3185.

Fu, L. and C. L. Kane, 2009 Josephson current and noise at a superconductor/quantum-spin-hall-insulator/superconductor junction. Phys. Rev. B **79**: 161408.

Fujii, T., S. H. Matsuo, K. Takashima, N. Hatakenaka, S. Kurihara, *et al.*, 2009 Theoretical studies on dynamical casimir effect in a superconducting artificial atom. In *QFEXT09 (University Oklahoma, USA)*, edited by K. A. Milt and M. Bordag, p. 344, World Scientific.

Geva-zatorsky, N. *et al.*, 2006 Oscillations and variability in the p53 system. Mol. Syst. Biol. pp. 1–13.

Ghosh, P., S. Sen, S. S. Riaz, and D. S. Ray, 2011 Controlling birhythmicity in a self-sustained oscillator by time-delayed feedback. Phys. Rev. E **83**: 36205.

González, H., H. Arce, and M. R. Guevara, 2008 Phase resetting, phase locking, and bistability in the periodically driven saline oscillator: Experiment and model. Phys. Rev. E **78**: 36217.

Grib, A. *et al.*, 2006 Synchronization of serial intrinsic josephson junction arrays on vicinal substrates. Supercond. Sci. Technol. **19**: S200–S204.

Gross, B. *et al.*, 2013 Modeling the linewidth dependence of coherent terahertz emission from intrinsic josephson junction stacks in the hot-spot regime. Phys. Rev. B **88**: 14524.

Haberichter, T., M. Marh, and R. Heinrich, 2001 Birhythmicity, trirhythmicity and chaos in bursting calcium oscillations. Biophys. Chem. **90**: 17–30.

Hadley, P., M. R. Beasley, and K. Wiesenfeld, 1988 Phase locking of josephson-junction series arrays. Phys. Rev. B **38**: 8712–8719.

Hounsgaard, B. Y. J., H. Hultborn, B. Jespersen, and O. Kiehn, 1988 Bistability of alpha-motoneurones in the decerebrate cat and in the acute spinal cat after intravenous 5-hydroxytryptophan. J. Physiol. **405**: 345–367.

Kadji, H. E., J. C. Orou, R. Yamapi, and P. Woafo, 2007 Nonlinear dynamics and strange attractors in the biological system. Chaos, Solitons & Fractals **32**: 862–882.

Kingni, S. T., A. Cheukem, A. C. Chamgoué, and F. T. Kamga, 2020 Analysis of josephson junction with topologically nontrivial barrier. Eur. Phys. J. B **93**: 143–147.

Kingni, S. T., G. F. Kuiate, V. K. Tamba, A. V. Monwanou, J. Bio, *et al.*, 2019 Analysis of a fractal josephson junction with unharmonic current-phase relation. Supercond. Nov. Magn. **32**: 2295–2301.

Kleiner, R., D. Koelle, F. Ludwig, and J. Clarke, 2004 Superconducting quantum interference devices. state of art and applications. In *Proceedings of the IEEE*, volume 92, p. 1534.

Likharev, K. K., 1986 *Dynamics of Josephson junction and circuits*. Gordon Breach, New York.

Lima, F. C. E., A. R. Moreira, L. E. Machado, and C. A. S. Almeida, 2021 Statistical properties of linear majorana fermions. Int. J. Quantum Chem. **121**: e26749.

McDermott, R., S. Lee, B. T. Haken, A. H. Trabesinger, A. Pines, *et al.*, 2004 Microtesla mri with a superconducting quantum interference device. Proc. Natl. Acad. Sci. **101**: 7857–7861.

Morita, M., K. Iwamoto, and M. Sen, 1989 Transition sequence and birhythmicity in a chemical oscillation model showing chaos. Phys. Rev. A **40**: 6592–6596.

Mourik, V., K. Zou, S. M. Frolov, S. R. Plissard, E. P. A. M. Bakkers, *et al.*, 2012 Signatures of majorana fermions in hybrid superconductor-semiconductor nanowire devices. Science **336**: 1003–1007.

Nana, B., S. B. Yamgoué, I. Kemajou, R. Tchitnga, and P. Woafo, 2018 Dynamics of an rlc series circuit with hysteretic iron-core inductor. Chaos, Solitons and Fractals **106**: 184–192.

Nori, F., 2008 Designing quantum-information-processing superconducting qubit circuits that exhibit lasing and other atomic-physics-like phenomena on a chip. In *APS March Meeting Abstracts*, pp. H15–001.

Oladapo, B. I., S. A. Zahedi, S. C. Chaluvadi, S. S. Bollapalli, and M. Ismail, 2018 Model design of a superconducting quantum interference device of magnetic field sensors for magnetocardiography. Biomed. Signal Process. Control **46**: 116–120.

Ovchinnikov, Y. N. and V. Z. Kresin, 2013 Networks of josephson junctions and their synchronization. Phys. Rev. B **88**: 214504.

Owen, C. S. and D. J. Scalapino, 1997 Inductive coupling of josephson junctions to external circuits. J. Appl. Phys. **2047**.

Ozyuzer, L. *et al.*, 2007 Emission of coherent thz radiation from superconductors. Science (80-.) **318**: 1291–1293.

Pountougnigni, O. V., R. Yamapi, G. Filatrella, and C. Tchawoua, 2019 Noise and disorder effects in a series of birhythmic josephson junctions coupled to a resonator. arXiv **3**: 1–13.

Ramakrishnan, B., A. A. Oumate, M. Tuna, Koyuncu, S. T. Kingni, *et al.*, 2022 Analysis, fpga implementation of a josephson junction circuit with topologically nontrivial barrier and its application to ring-based dual entropy core true random number generator. Eur. Phys. J. Spec. Top. **231**: 1049–1059.

Sosnovtseva, O. V., D. Setsinsky, A. Fausboll, and E. Mosekilde, 2002 Transitions between beta and gamma rhythms in neural systems. Phys. Rev. E **66**: 41901.

Stich, M., M. Ipsen, and A. S. Mikhailov, 2002 Self-organized pacemakers in birhythmic media. Phys. D Nonlinear Phenom. **171**: 19–40.

Veldhorst, M., C. G. Molenaar, C. J. M. Verwijs, H. Hilgenkamp, and A. Brinkman, 2012 Optimizing the majorana character of squids with topologically nontrivial barriers. Phys. Rev. B **86**: 024509–1075.

Ventura, A. *et al.*, 2007 Restoration of p53 function leads to tumour regression in vivo. Nature **445**: 661–665.

Wang, H. B. *et al.*, 2010 Coherent terahertz emission of intrinsic josephson junction stacks in the hot spot regime. Phys. Rev. Lett. **105**: 57002.

Wen, X.-G., 2006 Topological orders and edge excitations in fractional quantum hall states. Adv. Phys. **44**: 405–473.

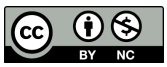
Wendin, G., 2017 Quantum information processing with superconducting circuits: a review. Reports Prog. Phys. **80**: 1060011–50.

Xie, H.-Y. and A. Levchenko, 2019 Topological supercurrents interaction and fluctuations in the multiterminal josephson effect. Phys. Rev. B **99**: 0945191–9.

Yamapi, R. and G. Filatrella, 2014 Noise effects on birhythmic

josephson junction coupled to a resonator. Phys. Rev. E **89**.
Yamapi, R., G. Filatrella, and M. A. Aziz-Alaoui, 2010 Global stability analysis of birhythmicity in a self-sustained oscillator. Chaos **20**: 1–12.
Yamapi, R., G. Filatrella, M. A. Aziz-Alaoui, and H. A. Cerdeira, 2012 Effective fokker-planck equation for birhythmic modified van der pol oscillator. Chaos **22**: 1–10.
Yue, X., W. Xu, L. Wang, and B. Zhou, 2012 Transient and steady-state responses in a self-sustained oscillator with harmonic and bounded noise excitations. Probabilistic Eng. Mech. **30**: 70–76.
Zakharova, A., T. Vadivasova, V. Anishchenko, A. Koseska, and J. Kurths, 2010 Stochastic bifurcations and coherence like resonance in a self-sustained bistable noisy oscillator. Phys. Rev. E **81**: 11106.

How to cite this article: Metsebo, J., Abdou, B., Ngatcha, D. T., Ngongiah, I. K., Kuate, P. D. K., and Pone, J. R. M. Analysis of a resistive-capacitive shunted Josephson junction with topologically nontrivial barrier coupled to a RLC resonator. *Chaos and Fractals*, 1(1), 31-37, 2024.



ADBA Chaos and Fractals
Cutting-Edge Scientific Solutions

e-ISSN: 3023-8951
RESEARCH ARTICLE
Vol.1 / No.1 / 2024 / pp.38-41
https://doi.org/10.69882/adba.chf.2024075

Digital Implementation of Chaotic Systems Using Nvidia Jetson AGX Orin and Custom DAC Converter

Berkay Emin *,1 **and Mustafa Yaz** α,2

*Department of Electronics and Automation, Osmancık Omer Derindere Vocational College, Hitit University, Corum, 19500, Turkiye, αDepartment of Electrical and Electronics Engineering,Faculty of Engineering and Architecture, Yozgat Bozok University, 66900 Yozgat, Turkiye.

ABSTRACT Digital implementation of chaotic systems has many advantages. Chaotic systems realized on digital platforms offer higher flexibility and computational accuracy compared to traditional analog systems. This flexibility and accuracy are of great importance for engineering applications. Furthermore, digital solutions are usually more cost-effective, which is a great economic advantage. In this study, Nvidia Jetson AGX Orin board and 16-bit converter board are used for digital realization of chaotic systems. The Nvidia Jetson AGX Orin is a platform that stands out with its high processing power and energy efficiency. The 16-bit converter card stands out with its sensitive data processing capacity. Thanks to this technological infrastructure, chaotic system equations can be successfully solved in a digital environment and more advanced, flexible, reliable and cost-effective solutions are obtained. These solutions provide significant advantages in various applications in engineering. In particular, the use of digital chaotic systems in areas such as modeling, simulation and control of complex systems offers more efficient and effective results than traditional methods. Digital platforms also facilitate parametric investigations and optimization processes of chaotic systems, enabling more in-depth analysis.

KEYWORDS
Chaotic system
Nvidia Jetson AGX Orin
Digital analog converter
Oscilloscope

INTRODUCTION

With technological advancements, the implementation of chaotic systems in digital environments is becoming increasingly important. Powerful microcontrollers, such as the Nvidia Jetson AGX Orin board, provide an ideal platform for performing complex computations of these systems at high speed and efficiency. The use of 16-bit based converter boards, in particular, enables more precise and detailed results. Such advancements enhance the real-time application of chaotic systems, facilitating the creation of more secure communication systems.

In this context, Wang et al. have implemented a fractional-order Chua circuit system using a microcontroller-based circuit. In their study, the dynamic properties of the fractional-order Chua circuit system were calculated using the ATmega328p microcontroller on the Arduino UNO platform, and these calculations were transferred to a computer via USB (Wang *et al.* 2021).

Méndez-Ramírez *et al.* (2021) introduced a novel hyperchaotic system (NHS) derived from the Méndez-Arellano-Cruz-Martínez (MACM) 3D chaotic system. To achieve real-time simulation and execution of the NHS's digital version (DV), they utilized the Digital Signal Processing (DSP) capabilities of a 16-bit dsPIC microcontroller in conjunction with two external dual digital-to-analog converters (DACs). This configuration facilitates the conversion of digitally processed data into analog signals, making them applicable for real-time scenarios.

In their work, Fan and Ding developed a microcontroller-based platform including components such as the TMS320F28335 DSP chip and 16-bit DAC to demonstrate the hardware implementation and geometric control of the polynomial chaotic map (Fan and Ding 2023). In the study Dong and Yang (2024), the dynamics, periodic trajectories and DSP (Digital Signal Processing) implementation of a new memristor-based 4D hyperchaotic system were performed. A D/A dual-channel converter (DAC8552) was used to convert the digital array into analog array. Finally, the analog signal was input to an oscilloscope to obtain an image of the hidden hyperchaotic attractor. In their study, Köse and Mühürcü (2017) focused on the realization of a digital chaotic oscillator using a low-cost microcontroller and chose to use a cost-effective microcontroller instead of many circuit elements to generate chaos signals. In this study, the algorithm for generating chaotic signals is realized using a PIC18F452 microcontroller and a DAC7611. The results are visualized on the oscilloscope screen in the form of 1D and 2D graphs. This method provides a cost-effective solution with less hardware requirements and facilitates the generation of chaotic signals. Chiu *et al.* (2019) conducted a study on designing

Manuscript received: 28 July 2024,
Revised: 30 July 2024,
Accepted: 30 July 2024.

[1]berkayemin@hitit.edu.tr (**Corresponding author**)
[2]mustafa.yaz@bozok.edu.tr

and implementing a jerk circuit through a hybrid analog-digital approach. This involved using a mix of analog components, such as operational amplifiers configured as integrators and various passive elements, alongside digital components, including microcontrollers and DAC/ADC boards.

The experimental findings demonstrate that the proposed hybrid circuit effectively replicates the dynamics of the jerk system.The implementation of chaotic systems in digital environments offers significant advantages across various application areas. For instance, chaotic encryption methods used for secure data transmission protect data from unauthorized access. These methods leverage the dynamic and unpredictable nature of chaotic systems, making it extremely difficult to break encryption keys. In this context, the security advantages provided by chaotic systems are particularly important in military and financial communications. The literature includes designs of chaos-based encryption and random number generators using embedded systems.

Murillo-Escobar *et al.* (2023) proposed a chaos-based cryptographic algorithm utilizing the Badola map to ensure the confidentiality of biomedical signals, such as ECG, in telemedicine applications. They implemented this algorithm in an embedded system based on a low-cost 12-bit microcontroller. The study demonstrates that the proposed chaotic cryptographic algorithm is robust against common attacks and can be effectively used in secure, low-cost embedded systems for telemedicine applications. Emin and Musayev (2023) developed an RGB image encryption algorithm utilizing Lorenz-Rossler chaotic systems for the Nvidia Jetson Nano and Xilinx PYNQ Z1 embedded systems. Their results indicate that the Nvidia Jetson Nano board performs encryption and decryption more rapidly than the Xilinx PYNQ Z1 board.

Murillo-Escobar *et al.* (2024) introduced two novel pseudorandom number generators (PRNGs) based on newly developed hyperchaotic maps. These PRNGs were initially implemented on the Renesas RA4M1 32-bit microcontroller. Pande and Zambreno introduced a new chaotic stream cipher derived from a modified logistic map, designed for real-time applications. They subsequently implemented this cipher in hardware on a Xilinx Virtex6 FPGA. The proposed approach is well-suited for embedded devices that have strict limitations on power consumption, hardware resources, and real-time performance (Pande and Zambreno 2013).

Charalampidis *et al.* (2024) proposed a real-time chaos-based speech encryption scheme based on STM32 microcontroller. They proved the effectiveness of the encryption scheme with various tests. Janakiraman *et al.* (2018) developed a chaos-based image encryption algorithm featuring lightweight design, which was optimized for a 32-bit microcontroller. The performance analysis reveals that this algorithm is appropriate for real-time embedded systems, offering reduced memory usage and improved efficiency.

The organization of this paper is as follows: Section 2 presents the mathematical framework of the Four-Wing System introduced by Khaled *et al.* (2024), along with the simulation results for time series and phase portrait. Section 3 focuses on capturing the time series and phase portrait of the Four-Wing System with an oscilloscope, employing the Nvidia Jetson AGX Orin and a DAC-based converter board. The paper concludes with a summary of the findings in the final section.

MATHEMATICAL MODEL OF FOUR-WING SYSTEM

The formula for the Four-Wing System proposed is given in Equation 1 (Khaled *et al.* 2024).

$$\begin{aligned} \dot{x} &= a(y-x)+yz \\ \dot{y} &= bx+cy-xz \\ \dot{z} &= -dz+xy+1 \end{aligned} \tag{1}$$

where x, y and z are state variables and a, b, c and d are control parameters. System 1 exhibits chaotic behavior for $[x_0, y_0, z_0] = [0.5, 0.2, 0.5]$ and $a = 10$, $b = -5$, $c = 10$ and $d = 30$. The x, y and z time series of the system are shown in Figure 1. Also, the $x-y, x-z$ and $y-z$ phase portraits are presented in Figure 2

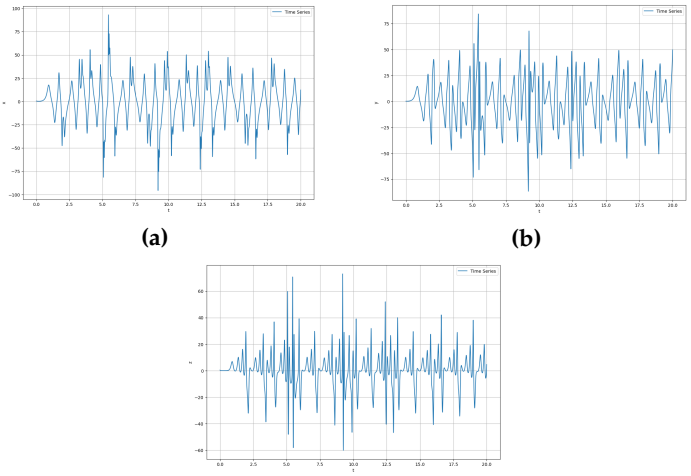


Figure 1 Time series of the Four-Wing System (a) x-t; (b) y-t; (c) z-t time series.

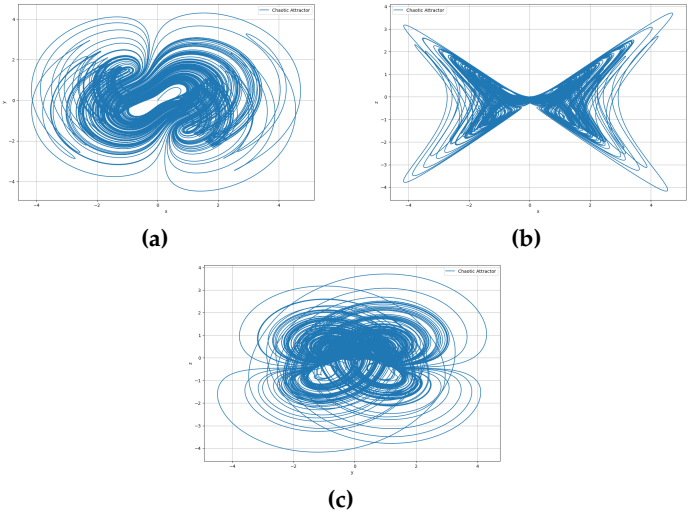


Figure 2 Phase portraits of the Four-Wing System (a)x-y; (b)x-z; (c)y-z phase portrait.

ELECTRONIC CIRCUIT IMPLEMENTATIONS

The high processing power of the Nvidia Jetson AGX Orin board used in this study is of great importance for modeling the complex nature of chaotic systems and performing real-time analysis on these models. In addition, the 16-bit based converter card offers high precision in converting analog signals to digital data, increasing the accuracy of the system and enabling this type of system to have a wider range of applications. The digital implementation of chaotic systems also offers significant advantages in the field of cryptography. The random and unpredictable nature of chaotic signals provides an ideal basis for secure communication systems. Digital platforms facilitate the generation and processing of these signals, contributing to the development of more secure cryptographic algorithms. Furthermore, chaotic implementations on digital systems allow parameters to be changed quickly and easily, enabling systems to adapt to various scenarios. In this chapter, the realization and digital implementation of the Four-Wing System on the Nvidia Jetson AGX Orin board is presented. All simulations and digital implementations of the Four-Wing System were performed using Python language on Nvidia Jetson AGX Orin embedded board and Jupiter programming environment. The block diagram of the application is given in Figure 3.

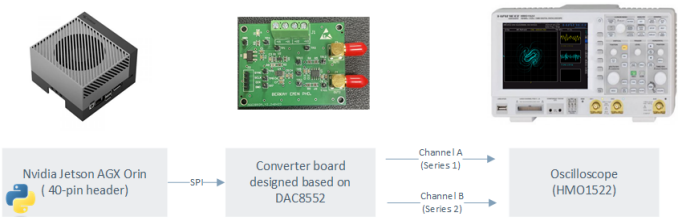


Figure 3 Block diagram of the application

As the Nvidia Jetson AGX Orin embedded board cannot provide analog output, the converter board designed by us was used. This converter board basically consists of the TI 8552 DAC and the OPA2197 opamp. The Nvidia Jetson AGX Orin embedded board was connected to this board via SPI (Serial Peripheral Interface). First, System 1 was analyzed and discretized using the Runge-Kutta 45 method on the Nvidia Jetson AGX Orin embedded board. Given that the converter board can output a maximum of ±5V, the values of the chaotic system were scaled to match this limit. Then, the obtained x, y and z chaotic values were transmitted to the designed converter board. After the board converted the output result, the oscilloscope output was obtained. The oscilloscope images of the time series and phase portraits of System 1 are shown in Figure 4 and Figure 5, respectively.

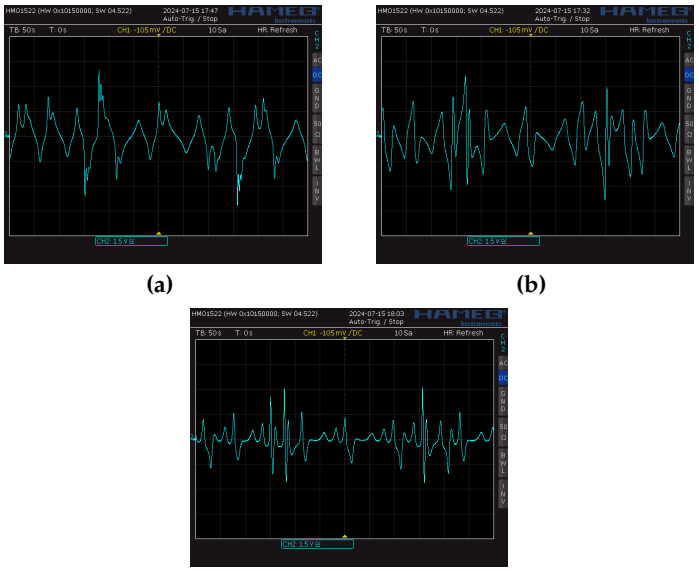


Figure 4 Oscilloscope Images of the time series of the Four-Wing System (a) x-t; (b) y-t; (c) z-t time series.

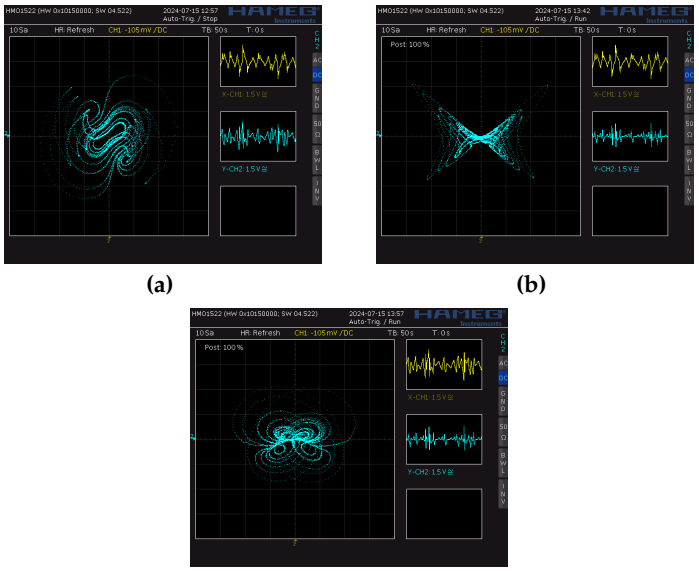


Figure 5 Oscilloscope Images of the time series of the Four-Wing System (a) x-t; (b) y-t; (c) z-t time series.

CONCLUSION

As a result, digital implementations of chaotic systems have great potential for flexibility, reliability and efficiency in engineering. Developed using the Nvidia Jetson AGX Orin board and 16-bit based converter board, these systems offer innovative solutions to complex engineering problems thanks to their high-performance computational capacity. In the future, such systems can be used for real-time signal encryption, random number generation and similar applications, which can further increase the effectiveness of engineering solutions. Furthermore, engineering applications of these chaotic systems can be used in various industries, which can open the door to sectoral innovations.

Availability of data and material

Not applicable.

Conflicts of interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Ethical standard

The authors have no relevant financial or non-financial interests to disclose.

LITERATURE CITED

Charalampidis, N., A. Iatropoulos, and C. Volos, 2024 Chaos based speech encryption using microcontroller. Integration **95**: 102128.

Chiu, R., D. López-Mancilla, C. E. Castañeda, O. Orozco-López, E. Villafaña-Rauda, *et al.*, 2019 Design and implementation of a jerk circuit using a hybrid analog–digital system. Chaos, Solitons Fractals **119**: 255–262.

Dong, C. and M. Yang, 2024 A novel 4D memristor-based hyperchaotic system with hidden attractors: Dynamics, periodic orbits analysis, and DSP realization. Chinese Journal of Physics **89**: 930–942.

Emin, B. and Z. Musayev, 2023 Chaos-based Image Encryption in Embedded Systems using Lorenz-Rossler System. Chaos Theory and Applications **5**: 153–159.

Fan, C. and Q. Ding, 2023 Design and geometric control of polynomial chaotic maps with any desired positive Lyapunov exponents. Chaos, Solitons Fractals **169**: 113258.

Janakiraman, S., K. Thenmozhi, J. B. B. Rayappan, and R. Amirtharajan, 2018 Lightweight chaotic image encryption algorithm for real-time embedded system: Implementation and analysis on 32-bit microcontroller. Microprocessors and Microsystems **56**: 1–12.

Khaled, B., R. Moghrabi, A. Sambas, S. Kacar, S. Uzun, *et al.*, 2024 A novel four-wing chaotic system with multiple equilibriums: Dynamical analysis, multistability, circuit simulation and pseudo random number generator (PRNG) based on the voice encryption. International Journal of Data and Network Science **8**: 989–1000.

Köse, E. and A. Mühürcü, 2017 Realization of a digital chaotic oscillator by using a low cost microcontroller. Engineering Review **37**: 341–348.

Méndez-Ramírez, R. D., A. Arellano-Delgado, M. A. Murillo-Escobar, and C. Cruz-Hernández, 2021 A new 4d hyperchaotic system and its analog and digital implementation. Electronics (Switzerland) **10**: 1–18.

Murillo-Escobar, D., C. Cruz-Hernández, R. M. López-Gutiérrez, and M. A. Murillo-Escobar, 2023 Chaotic encryption of real-time ECG signal in embedded system for secure telemedicine. Integration **89**: 261–270.

Murillo-Escobar, D., K. Vega-Pérez, M. A. Murillo-Escobar, A. Arellano-Delgado, and R. M. López-Gutiérrez, 2024 Comparison of two new chaos-based pseudorandom number generators implemented in microcontroller. Integration **96**: 102130.

Pande, A. and J. Zambreno, 2013 A chaotic encryption scheme for real-time embedded systems: Design and implementation. Telecommunication Systems **52**: 551–561.

Wang, J., L. Xiao, K. Rajagopal, A. Akgul, S. Cicek, *et al.*, 2021 Fractional-Order Analysis of Modified Chua's Circuit System with the Smooth Degree of 3 and Its Microcontroller-Based Implementation with Analog Circuit Design.

How to cite this article: Emin, B., and Yaz, M. Digital Implementation of Chaotic Systems Using Nvidia Jetson AGX Orin and Custom DAC Converter.*Chaos and Fractals*, 1(1), 38-41, 2024.