Intro to Chaos Final Assignment -

Chaotic dynamics in memristive circuits

By: Cristian Daescu and César González

This paper, "Chaotic dynamics in memristive circuits", explores the dynamic behavior of electrical circuits that incorporate memristors. Memristors, a combination of "memory" and "resistor," are fundamental electrical components whose resistance changes based on the electric charge that has flowed through them. Unlike a standard resistor with constant resistance, a memristor retains its past state, making it nonlinear.

Chaos refers to a system's sensitive dependence on initial conditions, where small changes in the starting state lead to vastly different outcomes. Although these systems are deterministic, their extreme sensitivity leads to unpredictability. The paper explores how memristors, with their inherent nonlinearity, contribute to the emergence of such complex dynamics in circuits.

Memristor's Influence on Circuit Dynamics

The paper examines how memristors influence the dynamics of various circuits, specifically focusing on three types:

- **Simple Series Circuit:** The analysis starts with a basic circuit of a capacitor, inductor, and memristor connected in series. Despite its simplicity, the circuit can generate both periodic (stable, repeating) and chaotic (complex, non-repeating) patterns, demonstrating the memristor's ability to introduce complexity.
- **Modified Chua Circuit:** The Chua circuit is known for exhibiting chaos, traditionally using a nonlinear negative resistance. Here, a memristor replaces this element, resulting in "coexistence of attractors" or multistability. This means the circuit can evolve to different stable states based on its initial conditions, showcasing the possibility of multiple distinct oscillations, both periodic and chaotic.
- **Modified Colpitts Circuit:** The Colpitts oscillator, typically used for high-frequency signals, normally employs a bipolar junction transistor (BJT). In this version, a memristor replaces the exponential nonlinear term of the BJT, leading to "extreme multistability". This results in a variety of behaviors including stable fixed points, different periodic cycles, and diverse chaotic patterns, influenced by initial conditions.

We will now reproduce the requested panels from the figures using the provided R code.

Task 1: Reproduce all the panels in Figure 2

This R code models the time-dependent behavior of a memristor by solving a system of ordinary differential equations (ODEs) that describe the evolution of the device's internal state variable w. This variable represents the normalized width of the doped region inside the memristor, which directly influences its electrical characteristics. The applied voltage is sinusoidal and defined as V <- v0 * sin(2 * pi * t / T), capturing a realistic input waveform for testing device response.

$$V = M \cdot I, \tag{1}$$

$$M = R_{\mathrm{ON}} \frac{w}{D} + R_{\mathrm{OFF}} \left(1 - \frac{w}{D}\right), \tag{2}$$

$$\begin{aligned} \dot{w} &= \eta \cdot f(w,p) \cdot I = \frac{\eta \cdot f(w,p) \cdot V}{M} \\ &= \frac{\eta \cdot f(w,p) \cdot v_0 \cdot \sin\left(\frac{2\pi t}{T}\right)}{R_{\mathrm{on}} \frac{w}{D} + \left(1 - \frac{w}{D}\right) \cdot R_{\mathrm{OFF}}}, \end{aligned} \tag{3}$$

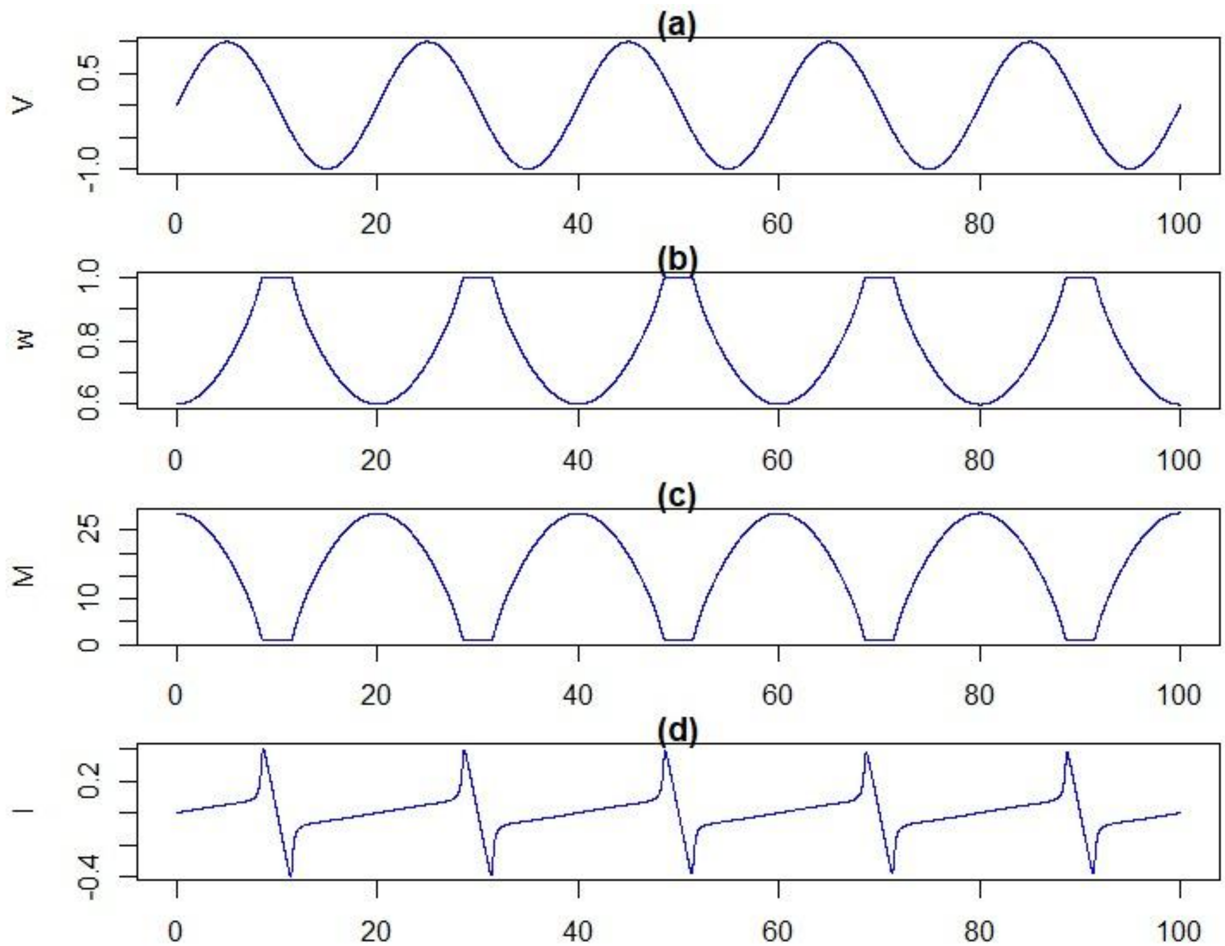
Table 1: Description and values of parameters [2].

Description	Parameter	Value
η	Polarity	1
D	Length of titanium diode memristor	1 nm
R_{OFF}	High resistance in low concentration dopant	70 Ω
R_{ON}	Low resistance in high concentration dopant	1 Ω
p	Integer value in the nonlinear function	10
v_0	Voltage Amplitude	1 V
T	Intrinsic period of oscillation	20 s
w_0	Initial device value	[0.0,1.0] nm

The memristor resistance M is computed as a weighted sum of low (R_ON) and high (R_OFF) resistance states based on w through M <- R_ON * w / D + R_OFF * (1 - w / D). The current I is then calculated via Ohm's law as I <- V / M. To reflect realistic boundary effects and nonlinear dopant drift, the code uses a window function f_wp defined by f_wp <- 1 - (2*w - 1)^(2*p), which smoothly reduces the rate of change of w near its physical limits of 0 and 1.

The state variable's time evolution follows the ODE dw_dt <- eta * f_wp * I, where eta is a polarity factor. The deSolve package numerically integrates this differential equation using an adaptive Runge-Kutta method. This solver adjusts the integration step size during runtime — decreasing it where the solution changes rapidly, and increasing it when the solution is smoother — thus balancing accuracy with computational efficiency. The solution is evaluated at fixed time points specified by time <- seq(0, 100, by = 0.01) over a total simulation time of 100 seconds.

In addition to the window function's effect, the code explicitly clamps w within the physically valid range using w <- min(max(w, 0), 1) at each step to avoid unphysical states. At every time step, the model computes and returns the current state, voltage V, resistance M, and current I. The final output is collected in a data frame and visualized via plots showing the temporal evolution of V, w, M, and I, effectively replicating the key features observed in Figure 2.



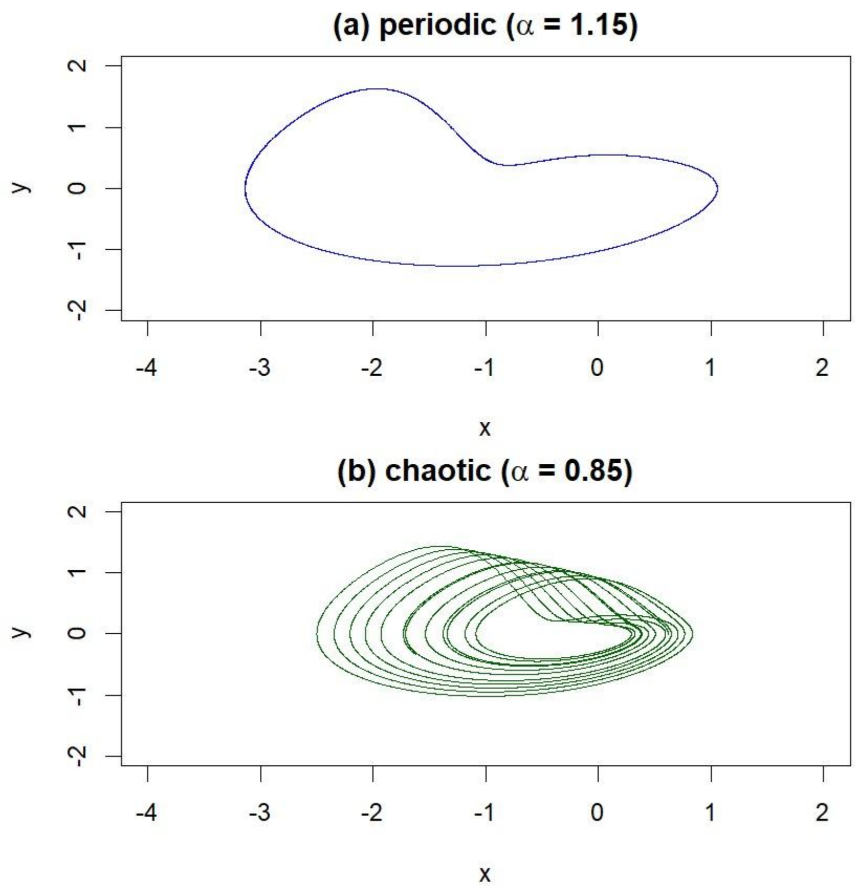
Task 2: Reproduce the phase portraits in Figure 5

This R code simulates the dynamics of a three-element series circuit that includes a capacitor, an inductor, and a memristor. The system is modeled by integrating ordinary differential equations (ODEs) using the **deSolve** library, based on the equations described in Section 3.1 of the paper. The **circuit_ode** function defines the three coupled ODEs for the state variables **x**, **y**, and **z**, representing the voltage across the capacitor, current through the inductor, and the internal memristive state, respectively. Specifically, the rates of change are calculated as **dx <- y / C**, **dy <- -1 / L * (x + beta * (z^2 - 1) * y)**, and **dz <- -y - alpha * z + y * z** .

Two parameter sets are prepared: **params_a** with **alpha = 1.15**, and **params_b** with **alpha = 0.85**, while capacitance **C**, inductance **L**, and nonlinear coefficient **beta** remain constant. Initial states are set as **state <- c(x = 0.1, y = 0, z = 0.1)**. The simulation runs over **time <- seq(0, 200, by = 0.01)**, totaling 200 seconds with fine time resolution.

Numerical integration is performed using the **ode** function with parameters **rtol=1e-8** and **atol=1e-8** for high precision. The solutions for both parameter sets are stored in **out_a** and **out_b** via calls like **ode(y = state, times = time, func = circuit_ode, parms = params_a, rtol=1e-8, atol=1e-8)**. To exclude early transient behavior and focus on the system's asymptotic dynamics, the first 50 seconds are discarded by filtering indices with **idx_a <- which(out_a$time > transient_time)** and similarly for **idx_b**.

Finally, phase portraits plotting **y** versus **x** are generated for both cases. The plot for **params_a** (with **alpha = 1.15**) shows a periodic attractor, visualized by **plot(out_a$x[idx_a], out_a$y[idx_a], type='l', col = "blue", ...)**. Conversely, the plot for **params_b** (**alpha = 0.85**) reveals a chaotic attractor, created through **plot(out_b$x[idx_b], out_b$y[idx_b], type='l', col = "darkgreen", ...)**. These visualizations reproduce Figure 5 in the paper, illustrating how varying **alpha** shifts the circuit from stable periodic oscillations to chaotic dynamics.

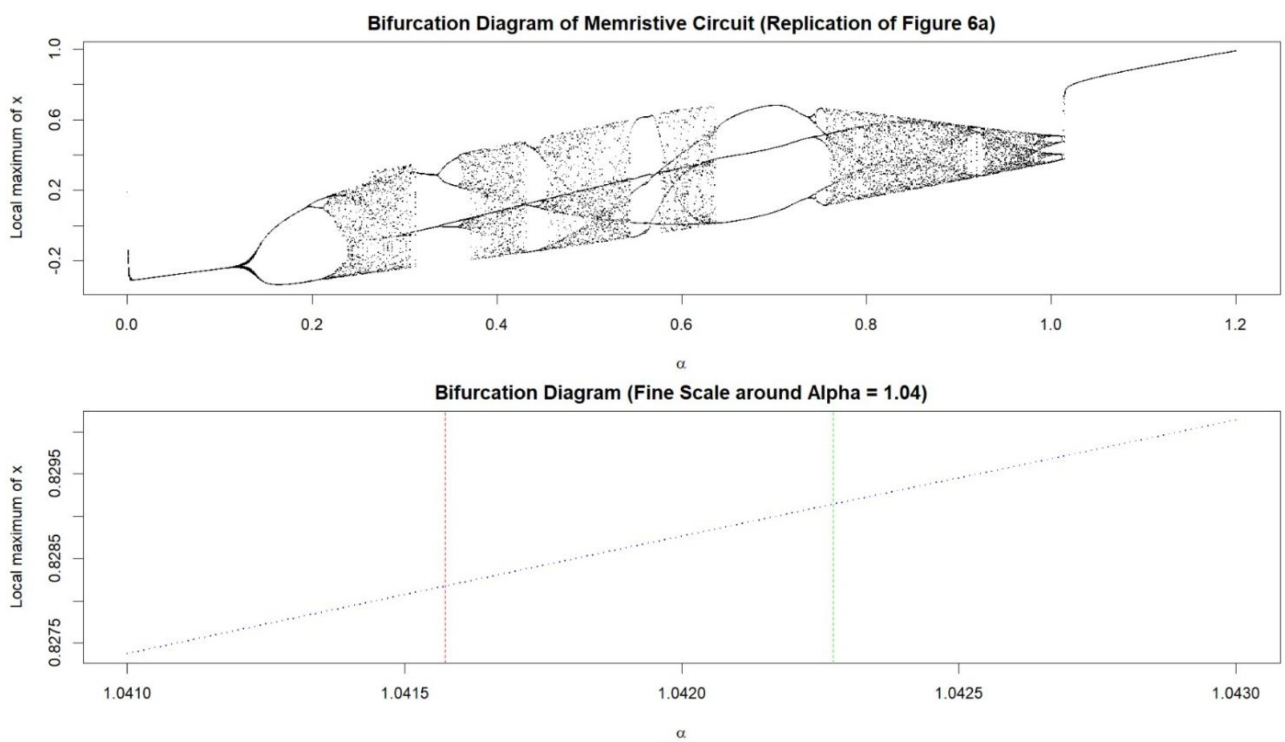


Task 3: Reproduce panel (a) of Figure 6

This code generates a bifurcation diagram of the memristive circuit's dynamics as a function of the parameter α, solving the same system of ODEs (memristor_circuit_ode) with constants C=1.2, L=3.3, and beta=1.34. The simulation scans α values from 0 to 1.2 in fine increments (alpha_values <- seq(0, 1.2, 0.001)), integrating over a long time window (0 to 500 seconds).

To construct the bifurcation diagram, the code detects upward crossings of the plane z=0 in the trajectory and records the corresponding x values (poincare_x_vals). Initial transients (e.g., first 300 seconds) are discarded to focus on steady-state behavior. For each α, the unique crossing points form the bifurcation plot's vertical slices showing periodic or chaotic regimes by variation in local maxima of x.

A detailed secondary analysis zooms around α≈1.041574, using very fine step sizes (alpha_fine_range <- seq(1.041, 1.043, 0.00001)) to capture subtle bifurcations and transitions. This reveals how small changes in α induce qualitative shifts in system dynamics, marked by divergence lines on the plot indicating critical parameter values for chaos onset. The final plots replicate panel (a) of Figure 6, demonstrating the circuit's sensitivity and nonlinear complexity in response to α.



Conclusion

The article effectively highlights the impact of memristors on the dynamic behavior of electronic circuits. Thanks to their unique memory and nonlinear characteristics, memristors are key in generating chaotic oscillations and showcasing multistability. By exploring simple RLC circuits, the Chua circuit, and the Colpitts oscillator, all modified with a memristor, the study offers evidence of the rich dynamic behaviors these devices can produce. This research enhances our understanding of nonlinear dynamics in electrical systems and points to the potential of memristors in future applications utilizing such complex phenomena.