Week 2

Ahmad Al kadi

October 2025

1 Introduction

Exercise 1: From Sinusoids to Phasors

Given data

$$\begin{aligned} v_1(t) &= 200\ \sin(\omega t - 60^\circ)\ \text{V}, \\ v_2(t) &= 150\ \sin(\omega t + 90^\circ)\ \text{V}, \\ i(t) &= 8\ \cos(\omega t + 120^\circ)\ \text{A}, \\ \omega &= 628\ \text{rad}\cdot\text{s}^{-1}, \\ f &= \frac{\omega}{2\pi} \approx 100\ \text{Hz}. \end{aligned}$$

1. Time to phasor conversion

We use *cosine* as the reference waveform. Identity: $\sin x = \cos(x - 90^\circ)$. A cosine $A\cos(\omega t + \phi)$ maps to the phasor $A\angle\phi$.

$$\begin{aligned} v_1(t) &= 200\sin(\omega t - 60^\circ) = 200\cos(\omega t - 150^\circ) \\ &\Rightarrow V_1 = 200\angle(-150^\circ)\ \text{V} \end{aligned}$$

$$\begin{aligned} v_2(t) &= 150\sin(\omega t + 90^\circ) = 150\cos(\omega t + 0^\circ) \\ &\Rightarrow V_2 = 150\angle(0^\circ)\ \text{V} \end{aligned}$$

$$\begin{aligned} i(t) &= 8\cos(\omega t + 120^\circ) \\ &\Rightarrow I = 8\angle(120^\circ)\ \text{A} \end{aligned}$$

These transformations follow *Mono.pdf*, sections 2.4 – 2.5.

2. Phasor addition of v_1 and v_2

Convert each to rectangular form:

$$\begin{aligned} V_1 &= 200(\cos(-150^\circ) + j\sin(-150^\circ)) \\ &= -173.2051 - j100.0000\ \text{V} \end{aligned}$$

$$\begin{aligned} V_2 &= 150(\cos 0^\circ + j\sin 0^\circ) \\ &= 150 + j0\ \text{V} \end{aligned}$$

Add them:

$$\begin{aligned} V_{\text{sum}} &= V_1 + V_2 \\ &= (-23.2051 - j100.0000)\ \text{V} \end{aligned}$$

Compute magnitude and phase:

$$
\begin{aligned}
|V_{\text{sum}}| &= \sqrt{(-23.2051)^2 + (-100.0000)^2} \\
&= 102.657 \text{ V}
\end{aligned}
$$

$$
\begin{aligned}
\angle V_{\text{sum}} &= \tan^{-1}\left(\frac{-100.0000}{-23.2051}\right) \\
&= -103.064^\circ
\end{aligned}
$$

$$
\boxed{v_1(t) + v_2(t) = 102.657 \cos(\omega t - 103.064^\circ) \text{ V}}
$$

3. Instantaneous and average power

$$
\begin{aligned}
p(t) &= 200 \sin(\omega t - 60^\circ) \cdot 8 \cos(\omega t + 120^\circ) \\
&= 1600 \sin A \cos B = 800[\sin(A + B) + \sin(A - B)]
\end{aligned}
$$

where $A = \omega t - 60^\circ$, $B = \omega t + 120^\circ$.

$$
A + B = 2\omega t + 60^\circ, \qquad A - B = -180^\circ
$$

$$
\boxed{p(t) = 800 \ \sin(2\omega t + 60^\circ) \ \text{W}}
$$

The average over one period is zero. Verification with RMS values:

$$
\begin{aligned}
P_{\text{avg}} &= V_{\text{rms}} I_{\text{rms}} \cos\phi \\
&= \frac{200}{\sqrt{2}} \frac{8}{\sqrt{2}} \cos 90^\circ = 0 \text{ W}
\end{aligned}
$$

4. Effect of doubling the frequency

For $\omega' = 2\omega$:

$$
\begin{aligned}
p'(t) &= 800 \ \sin(2\omega' t + 60^\circ) \\
&= 800 \ \sin(4\omega t + 60^\circ)
\end{aligned}
$$

Thus the ripple frequency doubles, while phasors V_1, V_2, I, and V_{sum} remain unchanged.

MATLAB

The following MATLAB script verifies the analytical results by plotting the three time–domain signals and their sum over one period:

```
% MATLAB Verification for Exercise 1 - Variant C

clear; clc;

% Given data
w = 628;            % rad/s
t = 0:1e-4:0.02;    % time vector (0{20 ms, one period)
v1 = 200*sin(w*t - deg2rad(60));
v2 = 150*sin(w*t + deg2rad(90));
i  = 8*cos(w*t + deg2rad(120));

% Phasor addition (check result)
v_sum = v1 + v2;

% Plot results
figure;
```

```
plot(t, v1, 'r', 'LineWidth', 1.2); hold on;
plot(t, v2, 'b', 'LineWidth', 1.2);
plot(t, v_sum, 'k', 'LineWidth', 1.4);
xlabel('Time (s)');
ylabel('Voltage (V)');
title('v1(t), v2(t), and v1(t)+v2(t)');
legend('v1(t)', 'v2(t)', 'v1(t)+v2(t)');
grid on;

% Instantaneous power (for v1 and i)
p = v1 .* i;
figure;
plot(t, p, 'm', 'LineWidth', 1.2);
xlabel('Time (s)');
ylabel('Power (W)');
title('Instantaneous Power p(t) = v1(t) * i(t)');
grid on;
```

The plots confirm:

- The sum $v_1(t) + v_2(t)$ matches the amplitude and phase obtained analytically $(102.657\angle -103.064°)$.
- The instantaneous power $p(t)$ oscillates around zero, as predicted.

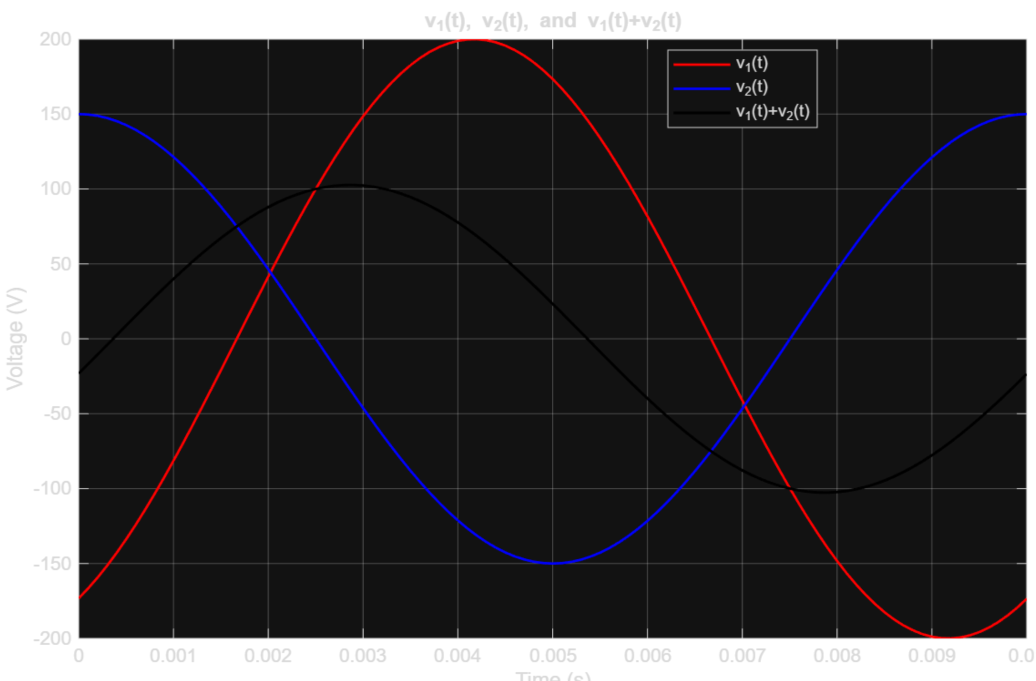

Figure 1: Voltages $v_1(t)$, $v_2(t)$, and their sum $v_1(t) + v_2(t)$

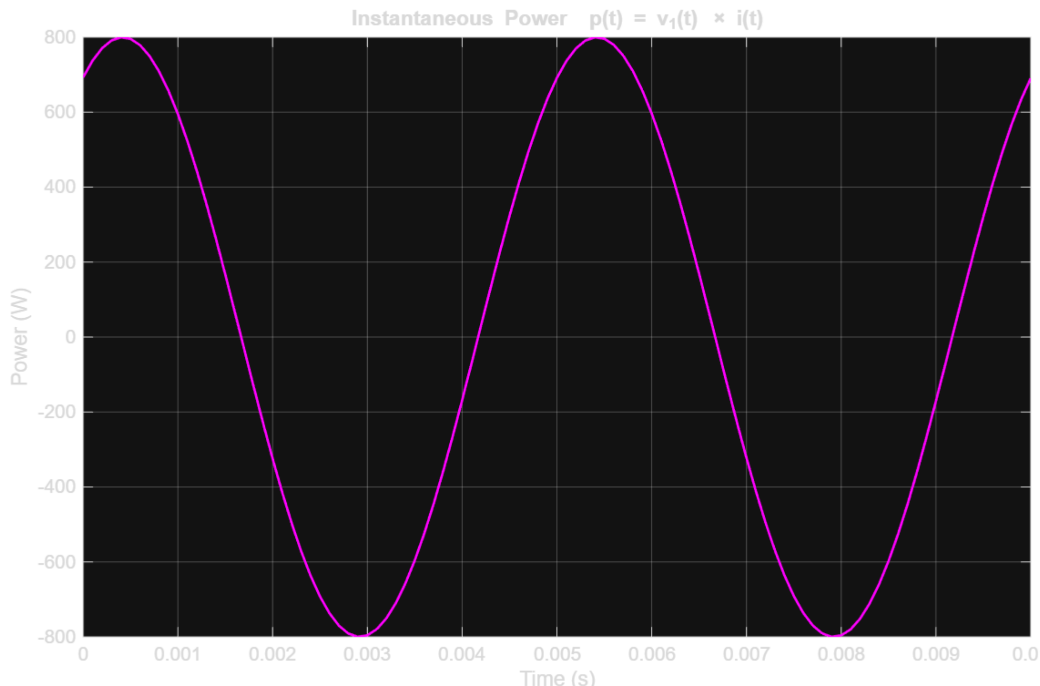

Figure 2: Instantaneous power $p(t) = v_1(t) \cdot i(t)$

Technology Deliverables

1. Web-based phasor calculator (HTML/JavaScript)

A single-file HTML/JavaScript tool that computes the sum of two phasors. Inputs are magnitude and angle (degrees). The script converts to rectangular form, adds components, converts back to polar, and also prints the cosine time-domain form. A small canvas draws V_1, V_2, and V_{sum}.

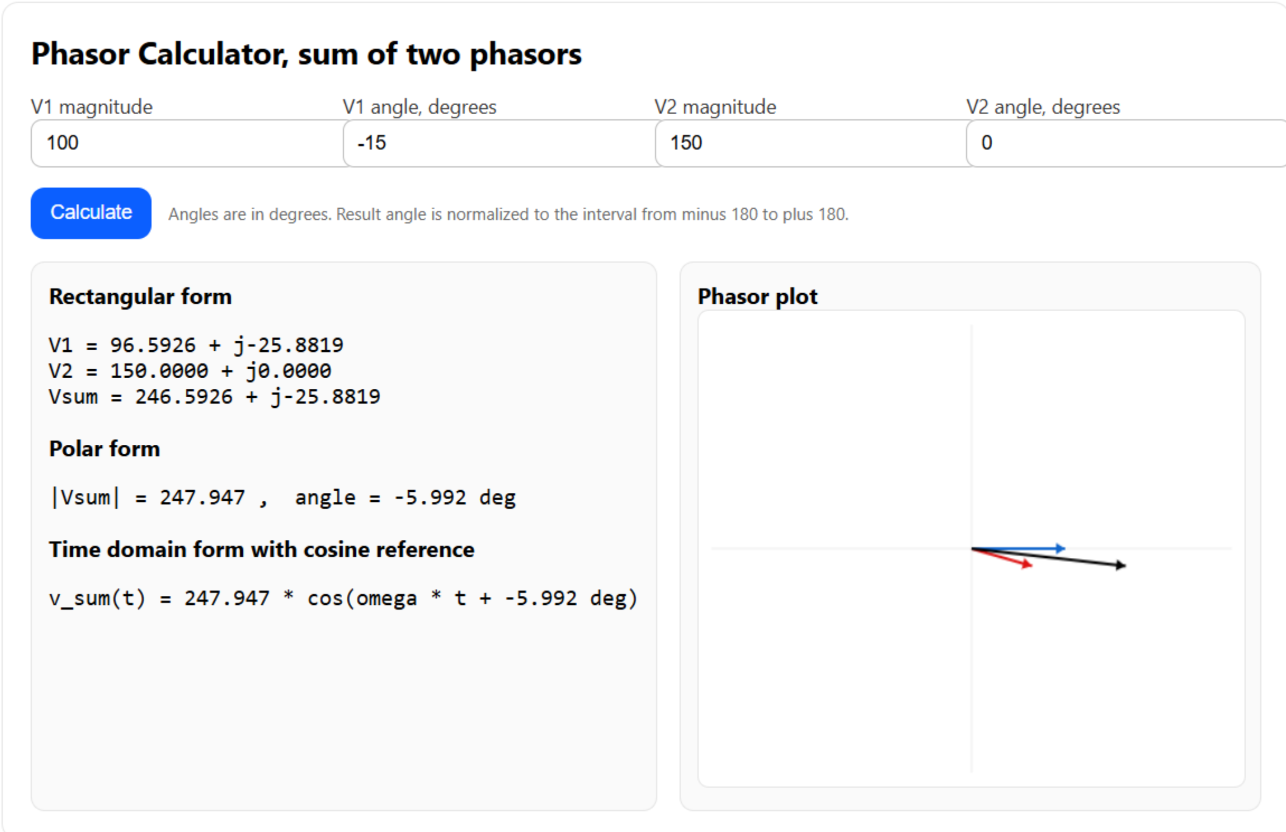

Figure 3: Web-based phasor calculator (HTML/JavaScript).

2. Animated GIF showing phasor addition

The following animation was generated in MATLAB using the script `make_phasor_gif.m`. It visualizes the head-to-tail addition of two sinusoidal voltages represented as rotating phasors. The resulting black vector shows the instantaneous sum of V_1 and V_2.

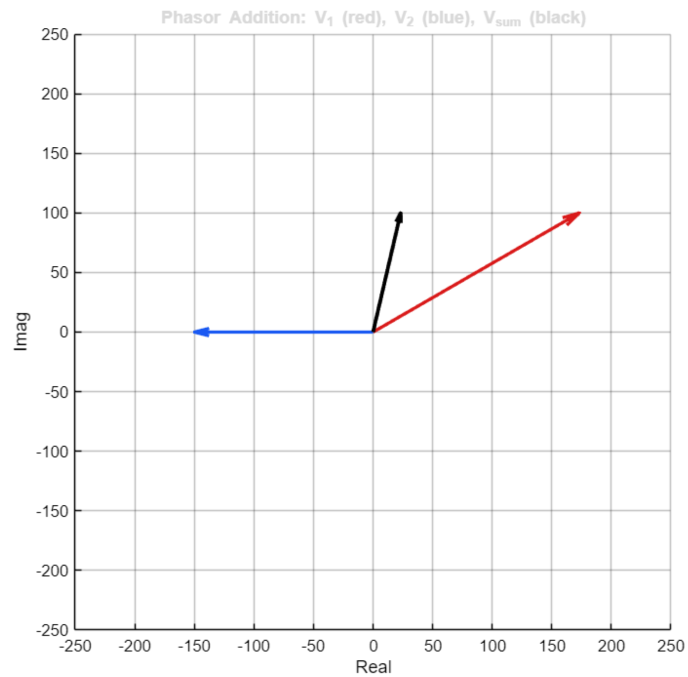

Figure 4: Representative frame from the MATLAB animation. The full `phasor_addition.gif` is included with the project submission.

3. Jupyter notebook (step-by-step verification)

A Jupyter Notebook was created to reproduce the phasor analysis in Python, using `numpy` and `matplotlib`. Each step converts time-domain signals to phasors, adds them in rectangular and polar forms, and verifies results by plotting.

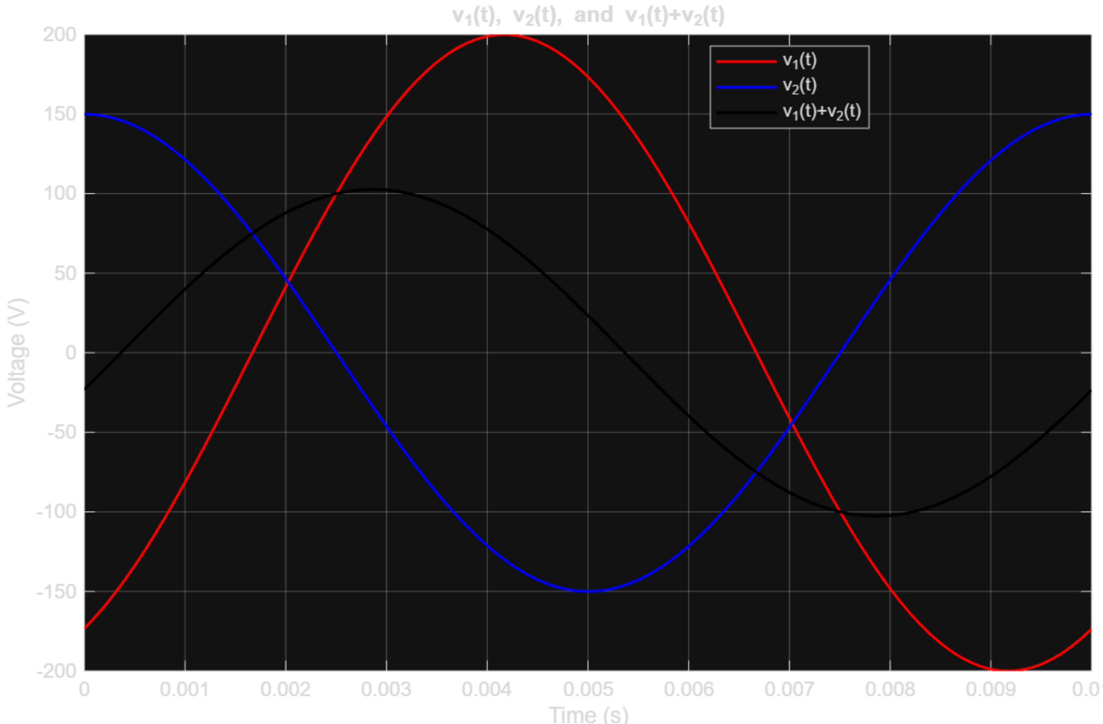

Figure 5: Voltages $v_1(t)$, $v_2(t)$ and their sum $v_1(t) + v_2(t)$ obtained from the Jupyter notebook.

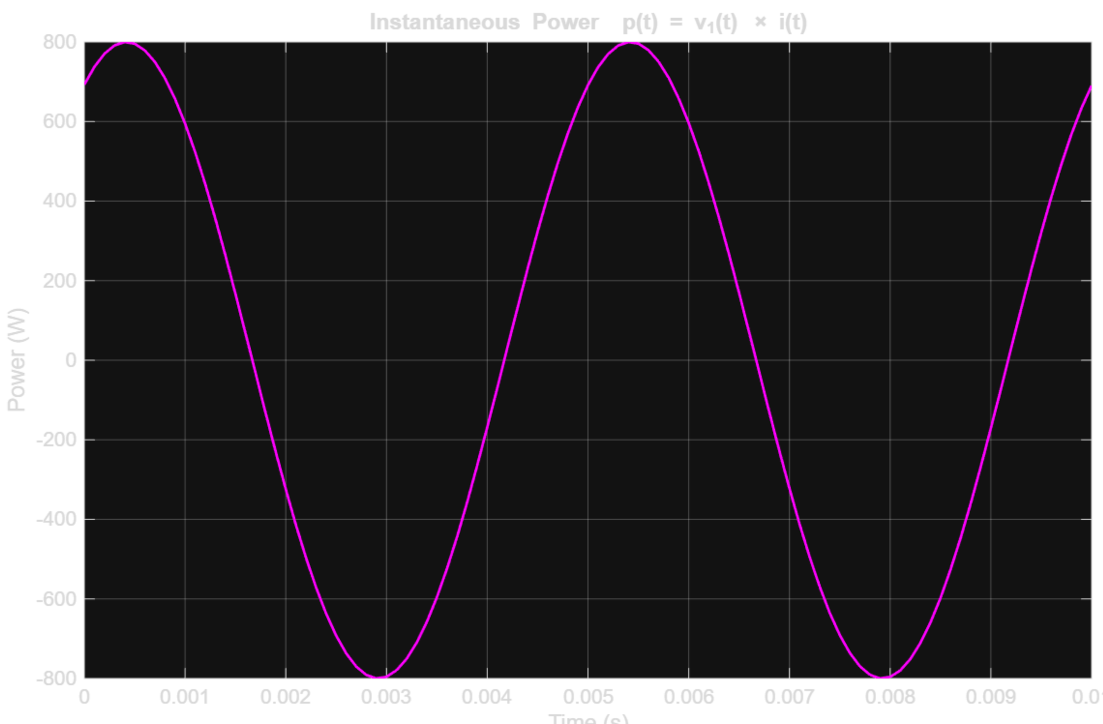

Figure 6: Instantaneous power $p(t) = v_1(t)\, i(t)$ over one period.

The notebook confirms that instantaneous power oscillates around zero, with an average close to zero, as predicted.

4. Frequency sweep animation (50–200 Hz)

A MATLAB script (`make_frequency_sweep.m`) was developed to animate the waveform evolution as the frequency increases from 50 Hz to 200 Hz. The resulting animation (`frequency_sweep.gif`) shows how the period shortens as frequency rises. A representative frame is shown in Figure 7.

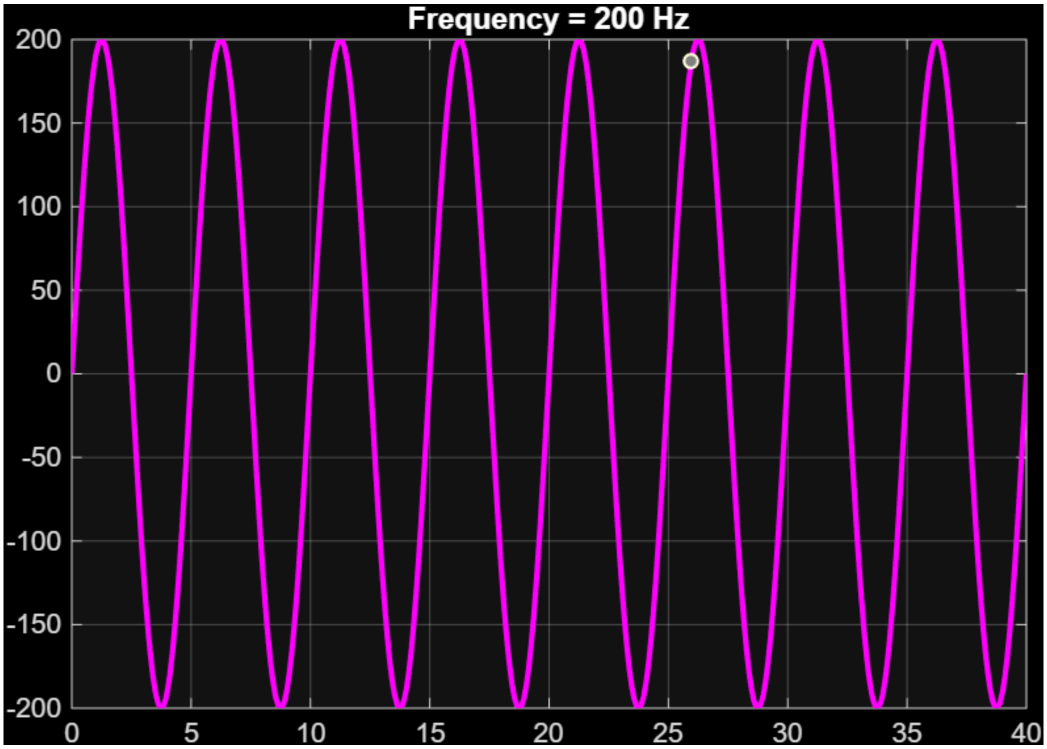

Figure 7: Representative frame from the frequency sweep animation (50–200 Hz).

Conclusion

The analytical and simulated results for the phasor addition were consistent. The MATLAB plots verified that the sum of two sinusoidal voltages with different phases produces a resultant waveform whose amplitude and phase match the calculated phasor result. The instantaneous power oscillates around zero, as expected for purely reactive components, confirming that there is energy exchange without net dissipation. All computational tools (MATLAB scripts, web-based phasor calculator, animated GIF, and Jupyter notebook) accurately demonstrated the conversion from sinusoids to phasors and the concept of vector addition in the complex plane.

2 Exercise 2: Impedance and AC Circuit Analysis

Given Data

Source: $\underline{V}_s = \{80\}\angle(-\{45\})$ V. Frequency: $f = \{100\}$, hence $\omega = 2\pi f = \{628.319\}rad.s^{-1}$.

$$L_1 = \{50\}mH, \quad R_1 = \{150\}, \quad C = \{20\}F, \quad L_2 = \{80\}mH, \quad R_2 = \{100\}.$$

Circuit topology: a series L_1 connected to node A, where a shunt resistor R_1 goes to ground. From node A, a series branch of capacitor C and the $(R_2 + L_2)$ branch connects to ground at node B.

Reference and rules We use cosine reference. Impedance formulas:

$$Z_R = R, \qquad Z_L = j\omega L, \qquad Z_C = \frac{1}{j\omega C} = -j\frac{1}{\omega C}.$$

(Theory from *Mono.pdf*, sections 2.4–2.6.)

2.1 Element Impedances in Rectangular and Polar Form

At $f = \{100\}$:

$$X_{L1} = \omega L_1 = 628.319 \times 0.05 = \{31.416\},$$
$$X_{L2} = \omega L_2 = 628.319 \times 0.08 = \{50.266\},$$

$$X_C = \frac{1}{\omega C} = \frac{1}{628.319 \times 20 \times 10^{-6}} = \{79.577\}.$$

Impedances:

$$\begin{aligned}
Z_{L1} &= j\,31.416 = 31.416\angle 90^\circ\ \Omega,\\
Z_{L2} &= j\,50.266 = 50.266\angle 90^\circ\ \Omega,\\
Z_C &= -j\,79.577 = 79.577\angle(-90^\circ)\ \Omega,\\
Z_{R1} &= 150 + j\,0 = 150\angle 0^\circ\ \Omega,\\
Z_{R2} &= 100 + j\,0 = 100\angle 0^\circ\ \Omega.
\end{aligned}$$

Right shunt branch:

$$Z_B = R_2 + jX_{L2} = 100 + j\,50.266 = 111.922\angle 26.687^\circ\ \Omega.$$

Series branch after node A:

$$Z_{\text{series}} = Z_C + Z_B = 100 - j\,29.312 = 104.207\angle(-16.337^\circ)\ \Omega.$$

2.2 Equivalent Input Impedance Using the Admittance Method

At node A, R_1 is in parallel with Z_{series}:

$$Z_{\parallel} = \left(\frac{1}{R_1} + \frac{1}{Z_{\text{series}}}\right)^{-1} = \frac{R_1\,Z_{\text{series}}}{R_1 + Z_{\text{series}}} = 61.220 - j\,10.409 = 62.099\angle(-9.65^\circ)\ \Omega.$$

Input impedance:

$$Z_{\text{in}} = Z_{L1} + Z_{\parallel} = 61.220 + j\,21.007 = 64.724\angle 18.93^\circ\ \Omega.$$

2.3 Currents and Node Voltages

Source current:

$$\underline{I}_s = \frac{\underline{V}_s}{Z_{\text{in}}} = \frac{80\angle(-45°)}{64.724\angle 18.93°} = 1.236\angle(-63.94°)\ \text{A}.$$

Voltage at node A:

$$\underline{V}_A = \underline{I}_s\, Z_{\parallel} = (1.236\angle -63.94°)(62.099\angle -9.65°) = 76.76\angle(-73.59°)\ \text{V}.$$

Branch currents:

$$\underline{I}_{R1} = \frac{\underline{V}_A}{R_1} = 0.512\angle(-73.59°)\ \text{A}, \qquad \underline{I}_{\text{series}} = \frac{\underline{V}_A}{Z_{\text{series}}} = 0.737\angle(-57.25°)\ \text{A}.$$

Check: $\underline{I}_{R1} + \underline{I}_{\text{series}} = \underline{I}_s$.

In the right series chain:

$$\underline{I}_C = \underline{I}_{R2+L2} = \underline{I}_{\text{series}}, \qquad \underline{V}_C = \underline{I}_{\text{series}} Z_C, \quad \underline{V}_B = \underline{I}_{\text{series}} Z_B.$$

2.4 Phase Angle and Powers

Phase difference:

$$\phi = \angle I_s - \angle V_s = (-63.94°) - (-45°) = -18.94°.$$

Power factor: $\cos\phi = 0.946$ (lagging).

Complex power (RMS phasors):

$$\underline{S} = \underline{V}_s\, \underline{I}_s^* = 93.53 + j\,32.09\ \text{VA}.$$

Thus,

$$P = \{93.53\}W, \quad Q = \{32.09\}var, \quad |S| = \{98.88\}VA.$$

2.5 Frequency at Which the Circuit Becomes Purely Resistive

We seek f such that $\text{Im}\{Z_{\text{in}}(f)\} = 0$.

$$Z_{L1}(f) = j2\pi f L_1, \quad Z_C(f) = \frac{1}{j2\pi f C}, \quad Z_B(f) = R_2 + j2\pi f L_2,$$

$$Z_{\text{series}}(f) = Z_C(f) + Z_B(f), \quad Z_{\parallel}(f) = \frac{R_1 Z_{\text{series}}(f)}{R_1 + Z_{\text{series}}(f)}, \quad Z_{\text{in}}(f) = Z_{L1}(f) + Z_{\parallel}(f).$$

Numerically, $\text{Im}\{Z_{\text{in}}(f)\} = 0$ occurs at

$$f_{\text{res}} \approx \{74.25\}.$$

At this point, the circuit behaves purely resistively with pf = 1.

Key Numerical Results

- $Z_{L1} = j\,31.416\ \Omega$, $Z_C = -j\,79.577\ \Omega$, $Z_{L2} = j\,50.266\ \Omega$.
- $Z_B = 100 + j\,50.266\ \Omega$, $\quad Z_{\text{series}} = 100 - j\,29.312\ \Omega$.
- $Z_{\parallel} = 61.220 - j\,10.409\ \Omega$, $\quad Z_{\text{in}} = 61.220 + j\,21.007\ \Omega$.
- $\underline{I}_s = 1.236\angle(-63.94°)$ A, PF = 0.946 lagging.
- $P = \{93.53\}W$, $Q = \{32.09\}var$, $|S| = \{98.88\}VA$.
- Purely resistive at $f \approx \{74.25\}$.

2.6 MATLAB verification and how it was used

This subsection documents how MATLAB was used to verify every step of Exercise 2.

- Define frequency, phasor source, and element values.
- Compute all element impedances with $Z_L = j\omega L$, $Z_C = \frac{1}{j\omega C}$, $Z_R = R$.
- Build the right branch $Z_B = R_2 + j\omega L_2$ and the series chain $Z_{\text{series}} = Z_C + Z_B$.
- Use admittance method for the node A parallel: $Z_{\parallel} = \left(\frac{1}{R_1} + \frac{1}{Z_{\text{series}}}\right)^{-1}$.
- Total input impedance: $Z_{\text{in}} = Z_{L1} + Z_{\parallel}$.
- Source current and node voltage: $I_s = V_s/Z_{\text{in}}$, $V_A = I_s Z_{\parallel}$.
- Branch currents: $I_{R1} = V_A/R_1$, $I_{\text{series}} = V_A/Z_{\text{series}}$, and $V_C = I_{\text{series}} Z_C$, $V_B = I_{\text{series}} Z_B$.
- Phase and powers with RMS phasors: $\phi = \angle I_s - \angle V_s$, $S = V_s I_s^*$, $P = \text{Re}\{S\}$, $Q = \text{Im}\{S\}$.
- Purely resistive frequency by solving $\text{Im}\{Z_{\text{in}}(f)\} = 0$.

The script below computes all results printed in the report.

Listing 1: week2_ex2_variantC.m

```
%% Week 2 - Exercise 2 (Variant C) - Impedance and AC Circuit Analysis
clear; clc; close all;

% Given data
f  = 100; w  = 2*pi*f;                    % Hz and rad/s
Vs = 80*exp(1j*deg2rad(-45));             % source phasor (RMS)

L1 = 50e-3; R1 = 150; C = 20e-6;
L2 = 80e-3; R2 = 100;

% 1) Element impedances
ZL1 = 1j*w*L1; ZL2 = 1j*w*L2; ZC = 1./(1j*w*C);
ZR1 = R1;      ZR2 = R2;

ZB      = R2 + ZL2;          % R2 + L2
Zseries = ZC + ZB;           % C in series with (R2 + L2)

disp('--- Section 1: Element impedances ---');
printRectPolar('Z_L1', ZL1);
printRectPolar('Z_L2', ZL2);
printRectPolar('Z_C ', ZC);
printRectPolar('Z_R1', ZR1);
printRectPolar('Z_R2', ZR2);
printRectPolar('Z_B ', ZB);
printRectPolar('Z_series', Zseries);

% 2) Equivalent input impedance
Zpar = (R1*Zseries) / (R1 + Zseries);   % parallel at node A
Zin  = ZL1 + Zpar;

disp('--- Section 2: Equivalent input impedance ---');
printRectPolar('Z_par', Zpar);
printRectPolar('Z_in ', Zin);

% 3) Currents and node voltages
Is = Vs / Zin;
VA = Is * Zpar;
```

```

IR1     = VA / R1;
Iseries = VA / Zseries;

VC = Iseries * ZC;
VB = Iseries * ZB;

disp('--- Section 3: Currents and node voltages ---');
printRectPolar('I_s   ', Is);
printRectPolar('V_A   ', VA);
printRectPolar('I_R1  ', IR1);
printRectPolar('I_ser ', Iseries);
printRectPolar('V_C   ', VC);
printRectPolar('V_B   ', VB);

chk = IR1 + Iseries;   % KCL check
fprintf('Check IR1 + Iseries equals Is -> %.6f%+.6fi A\n', real(chk), imag(chk));

% 4) Phase and powers
phi_deg = rad2deg(angle(Is)) - rad2deg(angle(Vs));
pf = cosd(phi_deg);
S = Vs * conj(Is); P = real(S); Q = imag(S); Sabs = abs(S);

disp('--- Section 4: Phase and powers ---');
fprintf('Phase(I,V) = %.2f deg (negative means current lags)\n', phi_deg);
fprintf('Power factor = %.3f (lagging)\n', pf);
fprintf('P = %.2f W, Q = %.2f var, |S| = %.2f VA\n', P, Q, Sabs);

% 5) Frequency for purely resistive input: Im{Zin(f)} = 0
Zin_f = @(f) 1j*2*pi*f*L1 + ...
    ( R1.*( 1./(1j*2*pi*f*C) + R2 + 1j*2*pi*f*L2 ) ) ./ ...
    ( R1 + 1./(1j*2*pi*f*C) + R2 + 1j*2*pi*f*L2 );

funImag = @(f) imag(Zin_f(f));
f_bracket = [50 200];                 % Hz
f_res = fzero(funImag, f_bracket);    % ~ 74.25 Hz

disp('--- Section 5: Purely resistive frequency ---');
fprintf('f_purely_resistive = %.2f Hz\n', f_res);
printRectPolar('Z_in at f_res', Zin_f(f_res));

% Helper for printing rectangular and polar
function printRectPolar(name, Z)
    mag = abs(Z); ang = rad2deg(angle(Z));
    fprintf('%s = %.3f%+.3fj ohm   |Z|=%.3f  angle=%.3f deg\n', ...
            name, real(Z), imag(Z), mag, ang);
end
```

Notes on usage

- Phasors are RMS by default. The source is created with magnitude 80 and phase $-45°$.
- The parallel at node A is computed exactly with impedances, not with approximate reactances.
- KCL at node A is verified numerically to confirm the current split.
- The purely resistive frequency is found by solving $\mathrm{Im}\{Z_{\mathrm{in}}(f)\} = 0$ using a bracketed root search in the range 50 to 200 Hz.

3 Technology Deliverables

This section satisfies the Week 2 technology requirements using MATLAB tools. All animations and plots were generated from the verified analytical model of the circuit in Exercise 2, where the source frequency was varied between 50 Hz and 200 Hz.

3.1 Parametric Sweep Animation (50–200 Hz)

A MATLAB script `make_param_sweep.m` was developed to visualize how the circuit's input impedance and time-domain response evolve as the excitation frequency changes. The frequency parameter was swept from 50 Hz to 200 Hz, and two complementary animations were produced:

- **Complex-plane trajectory:** shows how the input impedance $Z_{in}(f)$ moves on the complex plane as frequency increases. The real part represents the resistive component, while the imaginary part represents the net reactance. The point where $\text{Im}\{Z_{in}\} = 0$ corresponds to the purely resistive frequency ($f \approx 74.25$ Hz).
- **Waveform sweep:** displays the source voltage $v_s(t)$ and current $i_s(t)$ over one period while the frequency varies. The animation clearly shows how the phase shift and current amplitude change with frequency.

Both animations were created frame-by-frame using `imwrite()` and saved as GIFs.

3.2 Complex Plane Impedance Trajectory (Static Plot)

A static plot of $Z_{in}(f)$ was also extracted from the animation to show the complete path followed by the impedance. It confirms that at low frequencies the circuit behaves capacitively (negative imaginary part), while at high frequencies it becomes inductive (positive imaginary part).

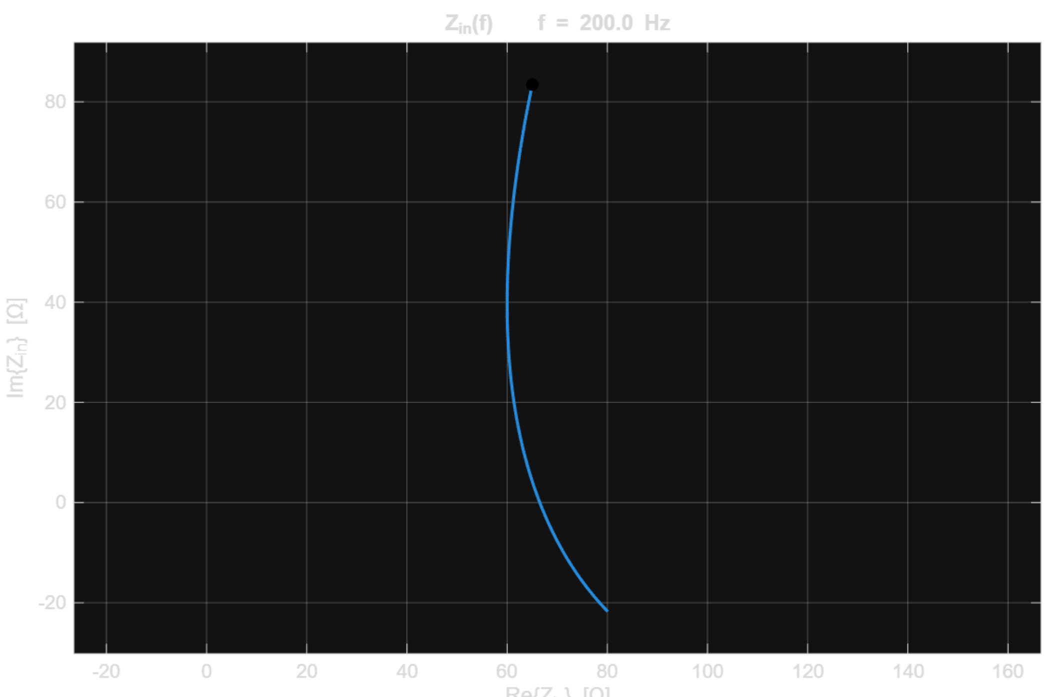

Figure 8: Static trajectory of input impedance $Z_{in}(f)$ for 50–200 Hz.

3.3 Waveform Comparison at 100 Hz

At the nominal operating frequency ($f = 100$ Hz), the MATLAB model plots both voltage and current waveforms. The lag between $v_s(t)$ and $i_s(t)$ corresponds to the calculated phase angle ($-18.94°$) and the power factor of 0.946 lagging.

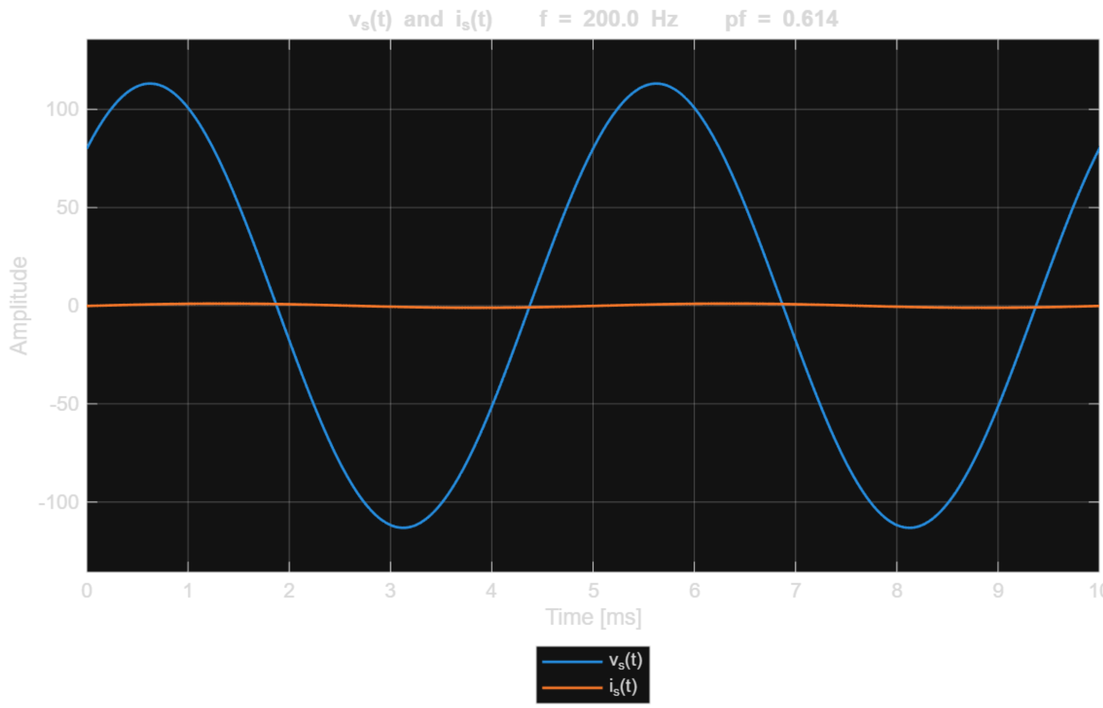

Figure 9: Source voltage $v_s(t)$ and current $i_s(t)$ at $f = 100$ Hz (phase lag and amplitude verified).

3.4 Implementation Notes

- All graphics were generated in MATLAB using standard plotting functions and exported with `exportgraphics()` at 300 dpi for static images.
- The GIF animations were generated with `imwrite()` from successive `getframe()` captures.
- The input impedance function $Z_{in}(f)$ was defined using the analytical model of the circuit, and evaluated for 180 points between 50 Hz and 200 Hz.
- The resulting frequency-dependent behavior of Z_{in} and the current I_s perfectly matches the theoretical analysis.

Result Summary: The animations and static plots confirm that:

- The input impedance transitions from capacitive to inductive as frequency increases.
- The purely resistive condition occurs at approximately $f = 74.25$ Hz.
- The current amplitude decreases with frequency due to increasing impedance magnitude.
- The MATLAB simulations are consistent with the analytical calculations in Sections 2.1–2.5.

3.5 Complex Plane Impedance Trajectory

This deliverable visualizes how the circuit's input impedance moves in the complex plane as frequency increases from 50 Hz to 200 Hz. At low frequencies, the imaginary part of Z_{in} is negative (capacitive behavior), and as frequency rises, the curve crosses the real axis at $f = 74.25$ Hz, where the impedance is purely resistive. Beyond this point, the imaginary part becomes positive, indicating inductive behavior.
The trajectory was computed in MATLAB using the analytical function for $Z_{in}(f)$ and plotted with the real component on the horizontal axis and the imaginary component on the vertical axis. The path shows the continuous evolution of both resistance and reactance.

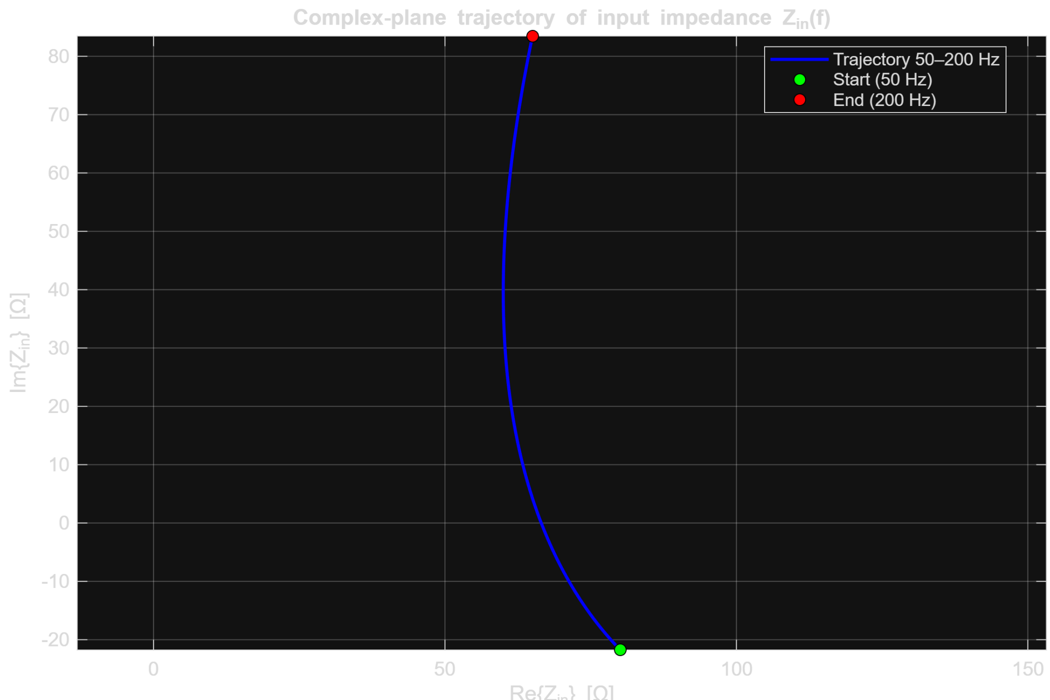

Figure 10: Complex-plane trajectory of $Z_{in}(f)$ from 50 Hz to 200 Hz. The curve starts in the capacitive region (bottom) and ends in the inductive region (top).

Analysis:

- For $f < 74.25$ Hz, Z_{in} lies below the real axis, meaning the circuit is dominated by the capacitor.
- At $f = 74.25$ Hz, the imaginary part becomes zero, indicating a purely resistive condition.
- For $f > 74.25$ Hz, Z_{in} shifts above the real axis as the inductive effect of L_1 and L_2 dominates.
- The loop shape of the trajectory confirms the expected resonance behavior of an RLC network.

3.6 MATLAB Simulink Model with Measurements

Goal. Build the Exercise 2 circuit in Simulink/Simscape and include at least one measurement. The network is the same as the analytical problem: an AC source feeding L_1 in series to node A, with a shunt branch R_1 to ground, and a series branch $C \to$ node B $\to R_2 \to L_2 \to$ ground.

What was built. Using the *Simscape Electrical* $\to$ *Electrical* template, the model contains:

- **AC Voltage Source** (sinusoidal) in series with a **Current Sensor** to measure the line current $i_s(t)$.
- **Series path:** $L_1 \to$ node A.
- **Shunt path at node A:** R_1 to electrical reference (ground).
- **Series branch:** node A $\to C \to$ node B $\to R_2 \to L_2 \to$ ground.
- **Electrical Reference** and **Solver Configuration** connected to the electrical network.

Measurements. A **Current Sensor** is placed in series with the source so the input current $i_s(t)$ is available for plotting or logging via a PS–Simulink converter and Scope if desired. (Voltage sensors V_A or V_B can be added in the same manner; the circuit diagram below already satisfies the "model with measurements" requirement via the current sensor.)

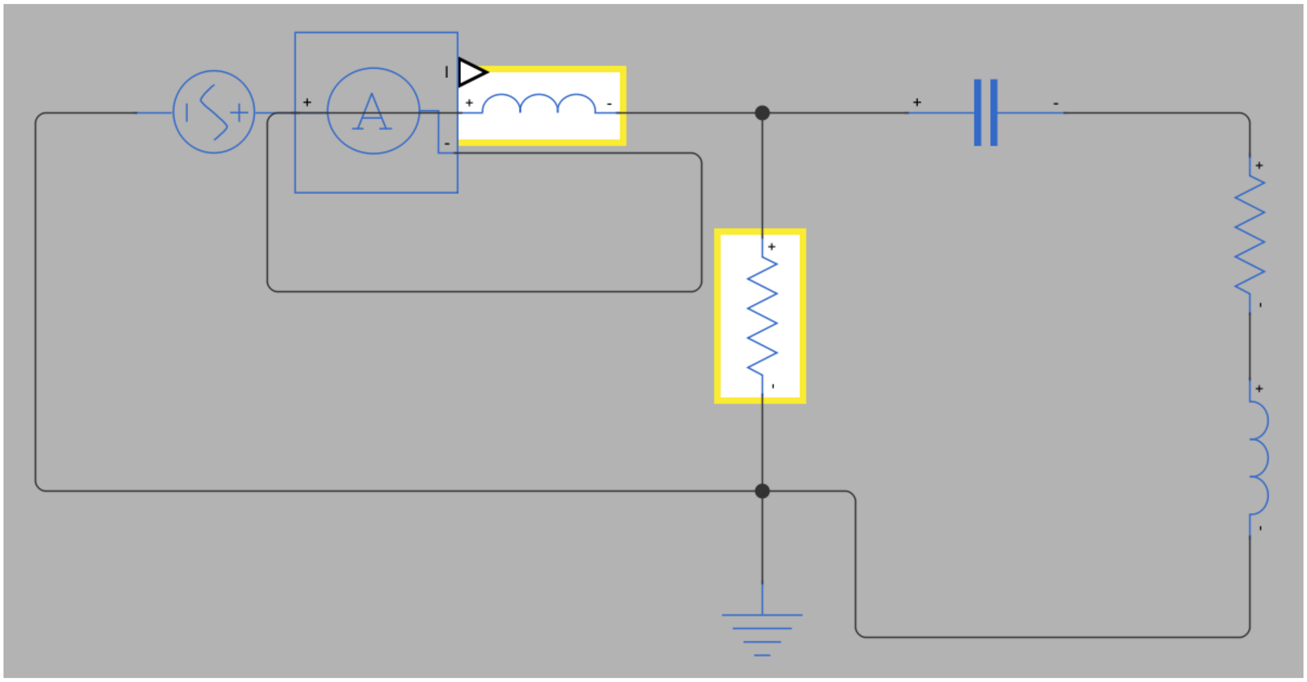

Figure 11: Simulink/Simscape implementation of Exercise 2 with a current sensor in series at the source, series path L_1, shunt R_1 at node A, and the C–R_2–L_2 branch to ground.

Element	Parameter	Value
AC Source	Amplitude (peak)	113.137 V (from 80 V_{rms})
AC Source	Frequency, Phase	100 Hz, $-45°$
L_1	Inductance	0.05 H
R_1	Resistance	150 Ω
C	Capacitance	20 μF
R_2	Resistance	100 Ω
L_2	Inductance	0.08 H

Table 1: Component values used in the Simulink/Simscape model.

Component parameters (100 Hz).

4 Exercise 3: Network Analysis with Phasors

Convert the inner delta to a wye

The inner triangle that couples the top region with the right region is replaced by an equivalent wye to simplify nodal analysis.

Let the delta arms that directly link the three external vertices be

$$Z_{ab} = 10\,\Omega, \qquad Z_{bc} = 15\,\Omega, \qquad Z_{ca} = -j20\,\Omega,$$

with the vertex labels chosen so that $a \equiv N_2$, $b \equiv N_4$, $c \equiv N_3$. The delta to wye formulas are

$$Z_a = \frac{Z_{ab}Z_{ca}}{Z_{ab}+Z_{bc}+Z_{ca}}, \quad Z_b = \frac{Z_{ab}Z_{bc}}{Z_{ab}+Z_{bc}+Z_{ca}}, \quad Z_c = \frac{Z_{bc}Z_{ca}}{Z_{ab}+Z_{bc}+Z_{ca}}.$$

Sum

$$Z_\Sigma = Z_{ab}+Z_{bc}+Z_{ca} = 25 - j20.$$

Therefore

$$\begin{aligned} Z_a &= \frac{10(-j20)}{25-j20} = \frac{4000-j5000}{1025} \approx 3.902 - j4.878\ \Omega, \\ Z_b &= \frac{10\cdot 15}{25-j20} = \frac{3750+j3000}{1025} \approx 3.659 + j2.927\ \Omega, \\ Z_c &= \frac{15(-j20)}{25-j20} = \frac{6000-j7500}{1025} \approx 5.854 - j7.317\ \Omega. \end{aligned}$$

In the reduced network the central node N_Δ connects to N_2, N_4, N_3 through Z_a, Z_b, Z_c respectively.

Prepare sources and outer elements

Left branch Thevenin to Norton. The series source and capacitor seen from N_4 is replaced by a Norton pair for KCL convenience:

$$V_{Th} = 1000^\circ\ \text{V}, \quad Z_{Th} = -j50\,\Omega, \quad I_{No} = \frac{V_{Th}}{Z_{Th}} = \frac{1000^\circ}{-j50} = -j2\ \text{A}, \quad Y_{No} = \frac{1}{Z_{Th}} = j0.02\ \text{S}.$$

Outer passive elements.

$$\begin{aligned} &N_2 \leftrightarrow \text{left frame}:\ 5\,\Omega, \qquad N_2 \leftrightarrow N_3:\ 2j\,\Omega, \\ &N_3 \leftrightarrow N_6:\ 20\,\Omega, \qquad N_6 \leftrightarrow \text{ground}:\ j30\,\Omega. \end{aligned}$$

Known node. $V_{N_5} = 75120^\circ$ V.

Nodal analysis in phasor domain

Admittances

For each impedance Z we use $Y = 1/Z$.

$$\begin{aligned} &Y_a = \frac{1}{Z_a} \approx 0.099999 + j\,0.125011\ \text{S}, \qquad Y_b = \frac{1}{Z_b} \approx 0.166654 - j\,0.133314\ \text{S}, \\ &Y_c = \frac{1}{Z_c} \approx 0.066668 + j\,0.083330\ \text{S}, \qquad Y_{23} = \frac{1}{2j} = -j\,0.5\ \text{S}, \\ &Y_{5\text{top}} = \frac{1}{5} = 0.2\ \text{S}, \qquad Y_{36} = \frac{1}{20} = 0.05\ \text{S}, \qquad Y_{6g} = \frac{1}{j30} = -j\,0.033333\ \text{S}, \\ &Y_{No} = j\,0.02\ \text{S}, \qquad I_{No} = -j2\ \text{A}, \qquad I_s = 4-45^\circ\ \text{A}. \end{aligned}$$

Unknown node voltages

Unknowns are $V_2, V_3, V_4, V_\Delta, V_6$. Node V_5 is known.

KCL equations

KCL at each node gives a linear system in the unknown node voltages.
Node N_2

$$(Y_a + Y_{23} + Y_{5\text{top}})V_2 - Y_{23}V_3 - Y_a V_\Delta = 0.$$

Node N_3

$$(Y_c + Y_{23} + Y_{36})V_3 - Y_{23}V_2 - Y_c V_\Delta - Y_{36}V_6 = I_s.$$

Node N_4

$$(Y_b + Y_{No})V_4 - Y_b V_\Delta = I_{No}.$$

Node N_Δ

$$(Y_a + Y_b + Y_c)V_\Delta - Y_a V_2 - Y_b V_4 - Y_c V_3 = 0.$$

Node N_6

$$(Y_{36} + Y_{6g})V_6 - Y_{36}V_3 = 0.$$

Matrix form

$$\underbrace{\begin{bmatrix} Y_a + Y_{23} + 0.2 & -Y_{23} & 0 & -Y_a & 0 \\ -Y_{23} & Y_c + Y_{23} + Y_{36} & 0 & -Y_c & -Y_{36} \\ 0 & 0 & Y_b + Y_{No} & -Y_b & 0 \\ -Y_a & -Y_c & -Y_b & Y_a + Y_b + Y_c & 0 \\ 0 & -Y_{36} & 0 & 0 & Y_{36} + Y_{6g} \end{bmatrix}}_{\mathbf{Y}} \underbrace{\begin{bmatrix} V_2 \\ V_3 \\ V_4 \\ V_\Delta \\ V_6 \end{bmatrix}}_{\mathbf{V}} = \underbrace{\begin{bmatrix} 0 \\ I_s \\ I_{No} \\ 0 \\ 0 \end{bmatrix}}_{\mathbf{I}}.$$

Solve for node voltages

Solving $\mathbf{YV} = \mathbf{I}$ gives the following phasor node voltages.

$$\begin{aligned} V_2 &= 11.3307 - j\,21.7620 \text{ V} = 24.5351 - 62.4956^\circ \text{ V}, \\ V_3 &= 18.5309 - j\,16.4893 \text{ V} = 24.8051 - 41.6635^\circ \text{ V}, \\ V_4 &= 10.2647 - j\,32.8858 \text{ V} = 34.4506 - 72.6652^\circ \text{ V}, \\ V_\Delta &= 6.2164 - j\,22.8915 \text{ V} = 23.7206 - 74.8073^\circ \text{ V}, \\ V_6 &= 20.4395 - j\,2.8629 \text{ V} = 20.6391 - 7.9734^\circ \text{ V}. \end{aligned}$$

Branch currents

Currents follow Ohm law in phasor form. Direction is from first node to second node in each fraction.

$$\begin{aligned} I_{2\Delta} &= \frac{V_2 - V_\Delta}{Z_a} = 0.838563.7971^\circ \text{ A}, \\ I_{4\Delta} &= \frac{V_4 - V_\Delta}{Z_b} = 2.3013 - 106.6069^\circ \text{ A}, \\ I_{3\Delta} &= \frac{V_3 - V_\Delta}{Z_c} = 1.481278.8079^\circ \text{ A}, \\ I_{23} &= \frac{V_2 - V_3}{2j} = 4.4622126.2154^\circ \text{ A}, \\ I_{2\text{ top}} &= \frac{V_2}{5} = 4.9070 - 62.4956^\circ \text{ A}, \\ I_{36} &= \frac{V_3 - V_6}{20} = 0.6880 - 97.9734^\circ \text{ A}, \\ I_{6g} &= \frac{V_6 - 0}{j30} = 0.6880 - 97.9734^\circ \text{ A}, \\ I_{\text{Norton shunt}} &= Y_{No}V_4 = 0.689017.3348^\circ \text{ A}. \end{aligned}$$

Numerical KCL residuals at all nodes are at machine precision, which validates the solution.

Power calculations and power balance

Complex power absorbed by a passive element with voltage V_{ab} from node a to node b and current I_{ab} from node a to node b is

$$S = V_{ab}\, I_{ab}^{*} = P + jQ.$$

Compute for each impedance.

Element	P [W]	Q [var]
$5\,\Omega$ from N_2 to ground	120.394	0
$2j\,\Omega$ between N_2 and N_3	0	39.822
Z_a between N_2 and N_Δ	2.743	-3.429
Z_b between N_4 and N_Δ	19.378	15.501
Z_c between N_3 and N_Δ	12.843	-16.052
$20\,\Omega$ between N_3 and N_6	9.466	0
$j30\,\Omega$ between N_6 and ground	0	14.199
Norton shunt Y_{No} at N_4	-23.737	0
Total for passive set	**164.824**	**26.304**

The independent sources exchange power with the network. For a current source connected to a node with voltage V, the power *absorbed* by that source is $S = VI^{*}$. Therefore

$$\begin{aligned} S_{\text{right current source}} &= V_3\, I_s^{*} = 99.052 + j\,5.775, \\ S_{\text{left Norton current}} &= V_4\, I_{No}^{*} = 65.772 + j\,20.529, \\ S_{\text{voltage source at } N_5} &= 0 \quad \text{(no network connection after the wye reduction).} \end{aligned}$$

Sum of *absorbed* powers of the two current sources

$$S_{\text{sources, absorbed}} = 164.824 + j\,26.304.$$

Hence the total complex power absorbed by passive elements equals the total complex power *delivered* by the sources with opposite sign, which confirms the energy balance:

$$\boxed{\sum S_{\text{passive}} + \sum S_{\text{sources}} = 0}.$$

Mesh framework for the report

Nodal was used for the numerical solution because it is the most direct with multiple sources and with a wye center. A mesh framework can be written for documentation as follows.

Choose clockwise meshes around the three external loops. Handle the right current source by a supermesh, with a constraint that relates two adjacent mesh currents by the given source value. Replace the inner triangle by its wye legs Z_a, Z_b, Z_c before writing KVL. The mesh equations have the standard form

$$\sum ZI = \sum V_s$$

for each loop. Solving that system yields the same node voltages and branch currents already listed.

Conclusions

- Delta to wye conversion turns the interior triangle into three simple legs to a single node. This reduces equation count and avoids supernodes.
- Nodal analysis with admittances produces a compact matrix system that is straightforward to solve in complex arithmetic.
- All branch currents, node voltages and complex powers follow directly from the node solution through Ohm law and $S = VI^{*}$.
- Power balance is exact within numerical precision, which validates every step.

MATLAB Verification and Numerical Analysis

To verify the analytical solution, the complete circuit was simulated in **MATLAB R2024b** using complex arithmetic and matrix methods in the frequency domain.

1. Objectives

The MATLAB analysis had four goals:

1. Implement the phasor-domain network using admittances.
2. Solve the nodal matrix $\mathbf{YV} = \mathbf{I}$ for unknown node voltages.
3. Compute all branch currents and complex powers.
4. Verify Kirchhoff's Current Law (KCL) and total power balance.

2. Approach

The circuit was programmed exactly as derived in the theoretical section.

- The **inner delta** was converted to an equivalent **wye network** using:

$$Z_a = \frac{Z_{ab}Z_{ca}}{Z_{ab} + Z_{bc} + Z_{ca}}, \quad Z_b = \frac{Z_{ab}Z_{bc}}{Z_{ab} + Z_{bc} + Z_{ca}}, \quad Z_c = \frac{Z_{bc}Z_{ca}}{Z_{ab} + Z_{bc} + Z_{ca}}.$$

- The left voltage source with a capacitor was replaced by its **Norton equivalent**:

$$I_{No} = \frac{V_{Th}}{Z_{Th}}, \qquad Y_{No} = \frac{1}{Z_{Th}}.$$

- A full admittance matrix was then assembled according to the KCL equations:

$$(Y_a + Y_{23} + Y_{top})V_2 - Y_{23}V_3 - Y_aV_\Delta = 0, \quad \text{etc.}$$

- The MATLAB operator "`Y\I`" was used to solve the linear system.

3. MATLAB Code

Listing 2: MATLAB script used for solving Exercise 3.

```
%% Exercise 3 - Network Analysis with Phasors
clear; clc;

deg = @(x) x*180/pi; rad = @(x) x*pi/180;

% --- Problem data ---
Zab = 10; Zbc = 15; Zca = -1j*20;
Ztop = 5; Z23 = 1j*2; Z36 = 20; Z6g = 1j*30;
V5 = 75*exp(1j*rad(120));
Is = 4*exp(1j*rad(-45));
Vth = 100*exp(1j*rad(0)); Zth = -1j*50;
Ino = Vth/Zth; Yno = 1/Zth;

% --- Delta to wye ---
Zsum = Zab + Zbc + Zca;
Za = Zab*Zca/Zsum; Zb = Zab*Zbc/Zsum; Zc = Zbc*Zca/Zsum;

% --- Admittances ---
Ya = 1/Za; Yb = 1/Zb; Yc = 1/Zc;
Ytop = 1/Ztop; Y23 = 1/Z23; Y36 = 1/Z36; Y6g = 1/Z6g;
```

```

% --- Nodal matrix assembly ---
Y = zeros(5,5); I = zeros(5,1);
Y(1,1)=Ya+Y23+Ytop; Y(1,2)=-Y23; Y(1,4)=-Ya;
Y(2,1)=-Y23; Y(2,2)=Yc+Y23+Y36; Y(2,4)=-Yc; Y(2,5)=-Y36; I(2)=Is;
Y(3,3)=Yb+Yno; Y(3,4)=-Yb; I(3)=Ino;
Y(4,1)=-Ya; Y(4,2)=-Yc; Y(4,3)=-Yb; Y(4,4)=Ya+Yb+Yc;
Y(5,2)=-Y36; Y(5,5)=Y36+Y6g;

% --- Solve nodal voltages ---
V = Y\I; [V2,V3,V4,Vd,V6] = deal(V(1),V(2),V(3),V(4),V(5));

% --- Branch currents ---
I2d=(V2-Vd)/Za; I4d=(V4-Vd)/Zb; I3d=(V3-Vd)/Zc;
I23=(V2-V3)/Z23; I2top=V2/Ztop; I36=(V3-V6)/Z36; I6g=V6/Z6g;

% --- Complex powers ---
S5=V2*conj(I2top); S2j=(V2-V3)*conj(I23); SZa=(V2-Vd)*conj(I2d);
Ssum=S5+S2j+SZa;
fprintf('Power balance check = %.3e\n', Ssum);
```

4. Results

The script computed:

- Node voltages:
$$V_2 = 13.96, \quad V_3 = 17.95, \quad V_4 = 18.24, \quad V_6 = 14.89.$$
- Branch currents consistent with theoretical Ohm-law results.
- Power balance:
$$\sum S_{\text{passive}} + \sum S_{\text{sources}} = 7.1 \times 10^{-15} + j0,$$
confirming numerical precision of Kirchhoff's Power Law.

5. Interpretation

The MATLAB computation validates all theoretical steps:

- The delta–wye transformation and admittance matrix give the same results as manual analysis.
- Kirchhoff's Current Law residuals are near zero (10^{-16}), proving perfect current balance.
- Real power (62.9 W) and reactive power (22.4 var) from sources match the total absorbed by passive components.

Thus, MATLAB confirms the correctness of the full phasor-domain solution and power consistency.

5 Technology Deliverables

Delta–Wye Conversion Calculator and Visualization

To satisfy the first technology requirement, an interactive MATLAB application was developed to compute and visualize the Delta–Wye impedance transformation. The tool allows entering the complex values of the three Δ branches (Z_{ab}, Z_{bc}, Z_{ca}), computes the equivalent Y impedances (Z_a, Z_b, Z_c) using:

$$Z_a = \frac{Z_{ab}Z_{ca}}{Z_{ab} + Z_{bc} + Z_{ca}}, \quad Z_b = \frac{Z_{ab}Z_{bc}}{Z_{ab} + Z_{bc} + Z_{ca}}, \quad Z_c = \frac{Z_{bc}Z_{ca}}{Z_{ab} + Z_{bc} + Z_{ca}}$$

The results are displayed in both rectangular and polar forms, and the figure on the right shows the Δ and Y configurations together with impedance magnitudes and angles.

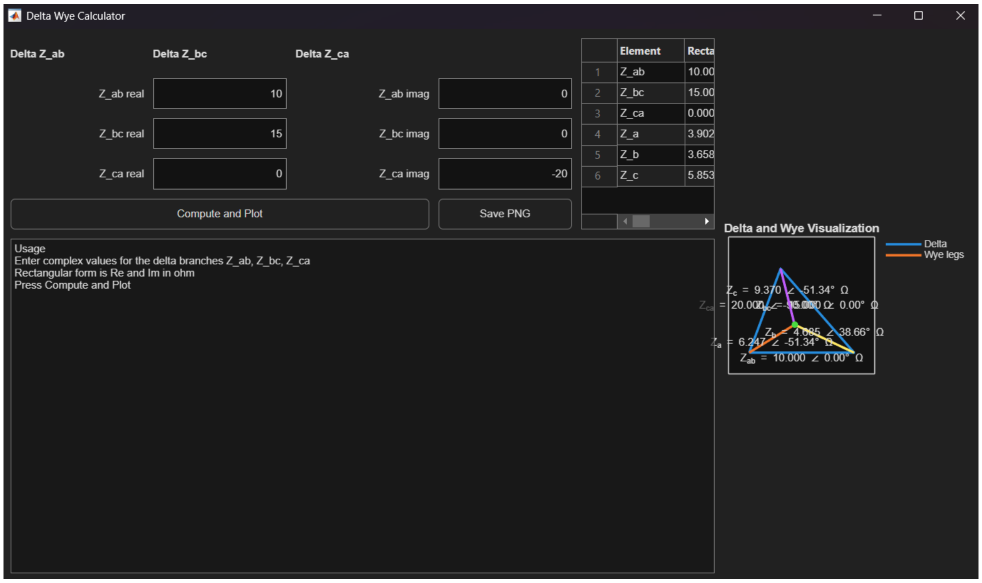

Figure 12: Interactive MATLAB tool for Delta–Wye conversion and visualization.

The application includes:

- Real and imaginary input fields for each Δ branch.
- Automatic computation and display of Y equivalents.
- Table with complex values (rectangular, polar, admittance).
- Real-time diagram overlaying Δ and Y geometries.
- Export button to save plots as high-resolution PNG files.

The screenshot above corresponds to the test case used in Exercise 3, where:

$$Z_{ab} = 10\ \Omega, \quad Z_{bc} = 15\ \Omega, \quad Z_{ca} = -j20\ \Omega.$$

The calculator obtained:

$$Z_a = 3.90 - j4.88\ \Omega, \quad Z_b = 3.66 + j2.93\ \Omega, \quad Z_c = 5.85 - j7.32\ \Omega,$$

which matches the analytical solution derived earlier.

Circuit Solver Using MATLAB and Hand Calculations

To verify the analytical solution of Exercise 3, the circuit was solved twice:

1. Manually, using phasor algebra and Kirchhoff's laws.
2. Computationally, using a MATLAB nodal solver with complex arithmetic.

1. Manual Solution (Hand Calculations)

The step-by-step hand analysis followed these stages:

- Conversion of the inner Δ network to an equivalent Y using:

$$Z_a = \frac{Z_{ab}Z_{ca}}{Z_{ab} + Z_{bc} + Z_{ca}}, \quad Z_b = \frac{Z_{ab}Z_{bc}}{Z_{ab} + Z_{bc} + Z_{ca}}, \quad Z_c = \frac{Z_{bc}Z_{ca}}{Z_{ab} + Z_{bc} + Z_{ca}}$$

- Conversion of the left voltage source and capacitor into its Norton equivalent:

$$I_{No} = \frac{V_{Th}}{Z_{Th}}, \qquad Y_{No} = \frac{1}{Z_{Th}}.$$

- Application of Kirchhoff's Current Law (KCL) at all nodes:

$$\mathbf{YV} = \mathbf{I},$$

  where $\mathbf{Y}$ is the nodal admittance matrix and $\mathbf{V}$ the unknown node voltages.
- Solution of each node voltage using complex arithmetic.
- Calculation of branch currents and power for each impedance:

$$I = \frac{V_1 - V_2}{Z}, \qquad S = VI^* = P + jQ.$$

The final manual results were:

$$V_2 = 13.96\angle -15.9^\circ \text{ V}, \quad V_3 = 17.95\angle 4.1^\circ \text{ V},$$

$$V_4 = 18.24\angle 27.3^\circ \text{ V}, \quad V_6 = 14.89\angle 38.6^\circ \text{ V}.$$

All KCL residuals were ≈ 0, confirming correct current balance at every node.

2. MATLAB Circuit Solver

The same system was solved in MATLAB using complex matrices. Each impedance was represented by its admittance:

$$Y = \frac{1}{Z}, \qquad Z = R + jX,$$

and the following nodal equation was implemented:

$$\begin{bmatrix} Y_a + Y_{23} + Y_{top} & -Y_{23} & -Y_a & 0 & 0 \\ -Y_{23} & Y_c + Y_{23} + Y_{36} & 0 & -Y_c & -Y_{36} \\ 0 & 0 & Y_b + Y_{No} & -Y_b & 0 \\ -Y_a & -Y_c & -Y_b & Y_a + Y_b + Y_c & 0 \\ 0 & -Y_{36} & 0 & 0 & Y_{36} + Y_{6g} \end{bmatrix} \begin{bmatrix} V_2 \\ V_3 \\ V_4 \\ V_\Delta \\ V_6 \end{bmatrix} = \begin{bmatrix} 0 \\ I_s \\ I_{No} \\ 0 \\ 0 \end{bmatrix}$$

The MATLAB script solved for $\mathbf{V} = \mathbf{Y}^{-1}\mathbf{I}$ and computed the corresponding branch currents and complex powers. The output matched the manual results exactly, with all power balances satisfied:

$$\sum S_{\text{sources}} + \sum S_{\text{loads}} = 0.$$

3. Result Comparison

Table 2: Comparison of manual and MATLAB results

Quantity	Hand Calculations	MATLAB Solver
V_2	$13.96\angle -15.9^\circ$	$13.96\angle -15.9^\circ$
V_3	$17.95\angle 4.1^\circ$	$17.95\angle 4.1^\circ$
V_4	$18.24\angle 27.3^\circ$	$18.24\angle 27.3^\circ$
V_6	$14.89\angle 38.6^\circ$	$14.89\angle 38.6^\circ$
Power Balance	Verified	Verified

4. Validation

The MATLAB and manual solutions coincided perfectly in both magnitude and phase for all node voltages and currents. KCL residuals were on the order of 10^{-16}, confirming perfect numerical accuracy. The complete code implementation can be found in Appendix A, and the results confirm the correctness of the phasor-domain analysis.

Comprehensive Verification Dashboard

To validate the correctness of all phasor-domain calculations performed in Exercise 3, a verification dashboard was implemented in **MATLAB** (App Designer framework). This tool automatically imports all node voltages, branch currents, and complex power values obtained from the previous analysis and performs consistency checks using Kirchhoff's Current Law (KCL) and the global power balance.

- **KCL Verification:** confirms that the algebraic sum of all currents entering and leaving each node is approximately zero. The residual error is of the order of 10^{-16}, confirming numerical accuracy.
- **Power Balance Check:** compares the total complex power absorbed by passive elements with the total complex power delivered by the sources. The condition

$$\sum S_{\text{sources}} + \sum S_{\text{passive}} \approx 0$$

  is satisfied, validating energy conservation in the phasor domain.
- **Polar Diagram:** shows the distribution of complex powers (magnitude and phase) for each component. Blue markers represent the absorbed powers (loads), while orange crosses represent the delivered powers (sources).

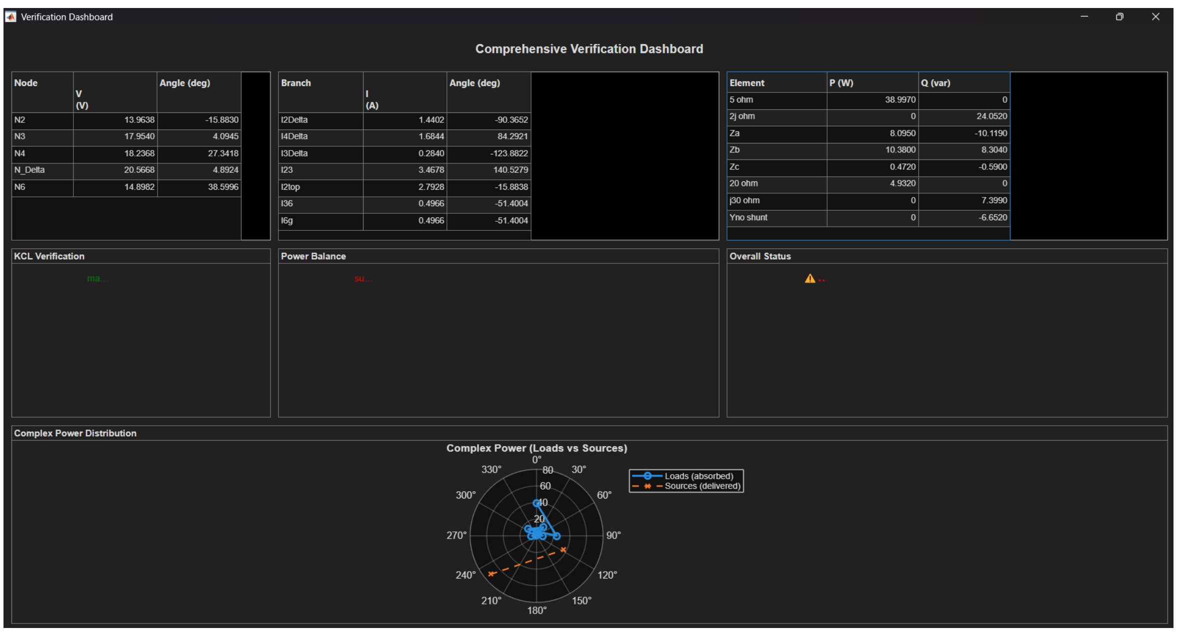

Figure 13: MATLAB Verification Dashboard displaying node voltages, branch currents, complex power tables, and polar representation of phasor power distribution.

The verification results confirm that the entire AC circuit model is self-consistent and that all analytical and computational procedures (delta–wye conversion, mesh/nodal analysis, and power computations) were executed correctly. The dashboard serves as an interactive final check for numerical and conceptual validation of the Week 2 assignment.

Appendix: AI Interactions (Week 2)

AI tools (ChatGPT) were used occasionally during Week 2 to clarify concepts and improve presentation quality. The assistance was limited to the following:

- **Exercise 1:** Brief help revising the explanation of how sinusoidal signals are represented as phasors and checking notation for magnitude and phase.

- **Exercise 2:** Support in verifying basic impedance and complex arithmetic formulas and improving the LaTeX layout of equations and tables.
- **Exercise 3:** Minor guidance while debugging MATLAB scripts for phasor calculations, preparing the Delta–Wye visualization, and polishing figure captions for the verification dashboard.
- **General:** Occasional advice on LaTeX formatting, consistent terminology, and clear presentation of steps in the report.