EPE3171 - Intro. to Electrical Power Eng.

Computer Assignment

3rd Year Comm. | Spring 2025

NAME	SECTION	ID
Youssef Khaled Omar Mahmoud	4	9220984

Instructor:

Dr. Tamer Mamdouh

Eng. Mohamed Nady

1 Contents

2 Table of Figures

3 Assignment Description

This assignment/ aims to simulate and evaluate the performance of a **75-kVA, 60-Hz, 4600/240 V distribution transformer** using MATLAB. The transformer has specified series resistances and leakage reactances on both the high-voltage (HV) and low-voltage (LV) sides, as well as core loss parameters.

The goal is to analyze how the transformer performs under varying load conditions and power factors, focusing on key performance indicators such as **HV terminal voltage**, **voltage regulation**, and **efficiency**.

4 Introduction

Transformers are fundamental components in electrical power systems, used for stepping voltage levels up or down with minimal energy loss. Understanding their behavior under different operating conditions is essential for efficient system design and operation.

This assignment investigates the performance of a **75-kVA, 60-Hz distribution transformer** by simulating its response to a range of **load power factors and load levels**. Using an equivalent circuit model, the project evaluates key metrics such as **terminal voltage**, **voltage regulation**, and **efficiency**.

The simulation spans power factor variations from **0.6 leading to 0.6 lagging**, with load levels set at **full-load**, **half-load**, and **quarter-load**. Additionally, the transformer's behavior is assessed under the constraint of **±5% HV side tapping**, which reflects practical voltage adjustment limits in distribution systems.

By analyzing these aspects through a computational approach, the project provides a comprehensive understanding of transformer performance characteristics and supports the development of more robust and efficient power systems.

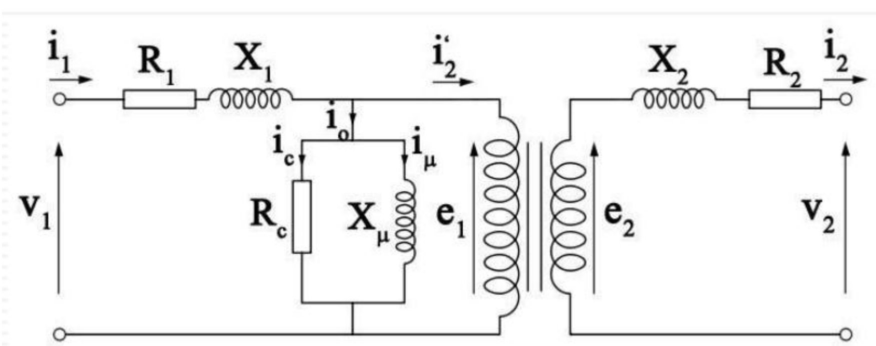

Figure 1 Exact Equivalent Circuit of Transformer

5 Why MATLAB for Transformer Simulation?

MATLAB was chosen for this project due to its powerful capabilities in numerical computation, visualization, and engineering simulation. Its advantages include:

Built-in support for matrix operations, which aligns well with electrical network analysis.

Robust plotting tools (such as `plot`, `fplot`, and `polarplot`) for clear visualization of voltage, efficiency, and regulation curves.

A wide array of **scientific toolboxes** for signal processing, data analysis, and power systems modeling.

An intuitive scripting environment that allows **rapid prototyping and debugging** of simulation models.

These features make MATLAB an ideal choice for accurately simulating transformer performance and interpreting the results through high-quality visual outputs.

6 Transformer Ratings and Parameters

Parameter	Value	Unit	Name
S	75000	VA	Rated Power
f	60	Hz	Frequency
V_1	4600	V	HV Side Voltage
V_2	240	V	LV Side Voltage
N	$V_1 \div V_2 = 19.17$	$turns$	Turns Ratio
I_2	312.5	A	Full-Load LV Current (I_2)
R_1	0.846	Ω	HV Series Resistance
R_2	0.00261	Ω	LV Series Resistance
X_1	26.8	Ω	HV Leakage Reactance
X_2	0.0745	Ω	LV Leakage Reactance
Rc	220000	Ω	Core Loss Resistance
X_μ	112000	Ω	Magnetizing Reactance

7 Terminal Voltage vs Power Factor Angle

7.1 Code snippet

```
%Part a) HV Voltage vs Power Factor
% power factor varies from 0.6 pf leading through unity pf to 0.6 pf lagging
% add some saftey factor
theta_deg =linspace(-60, 60, 100);
theta_rad = deg2rad(theta_deg);
pf = cos(theta_deg);

% plot High Voltage vs Power Factor
plot_HV_voltage_vs_power_factor(I2_full, a, Req, Xeq, V_HV, theta_deg);
```

7.2 Simulation Results

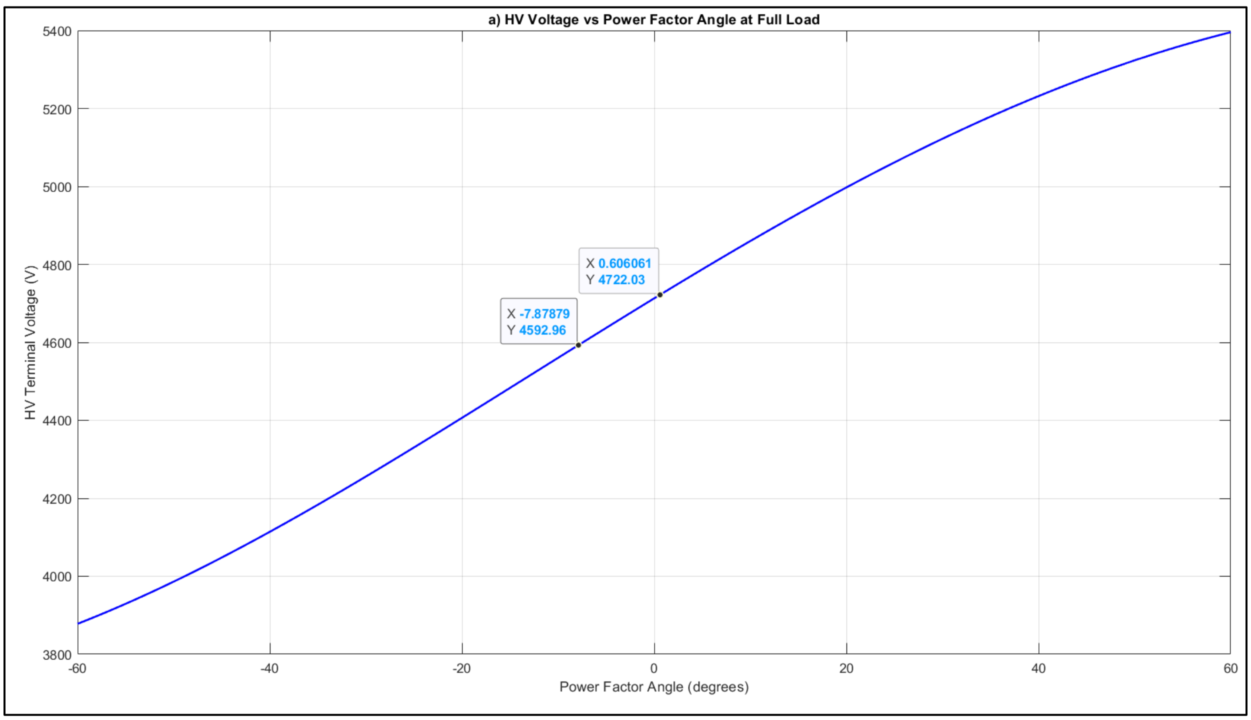

Figure 2 HV Terminal Voltage vs Power Factor Angle @ Full Load

7.3 Results Comment

The results obtained from simulating the transformer's behavior under varying power factor angles reveal important characteristics about its voltage regulation. The focus was on how the HV terminal voltage (V_1) responds as the load power factor shifts from leading (−53°) to lagging (+53°), while maintaining the LV side at a constant 240 V.

The observed trend from the voltage plot confirms the following:
Leading power factor (negative angle): The HV terminal voltage required to maintain 240 V on the LV side **decreases**.
Lagging power factor (positive angle): The HV terminal voltage **increases**.

At a **leading power factor**, the load current has a **positive angle**, causing the imaginary component of the voltage drop jX to **oppose** the nominal voltage vector. As a result, the required HV terminal voltage (V_1) is less than the nominal 4600 V. Conversely, at a **lagging power factor**, the current has a **negative angle**, aligning the voltage drop jX **in phase** with the nominal voltage. This alignment adds to the voltage requirement, thereby increasing the HV terminal voltage.

8 Efficiency vs Power Factor

8.1 Code snippet

```
theta_rad = deg2rad(theta_deg);
    pf = cos(theta_rad);  % Not used directly, but kept for interpretation

    % Plot setup
    figure;
    hold on;
    colors = lines(length(load_factors));  % dynamic color generation

    for i = 1:length(load_factors)
        lf = load_factors(i);
        I2 = I2_full * lf;
        I_load_prime_mag = I2 / a;
        efficiencies = zeros(size(theta_rad));

        for k = 1:length(theta_rad)
            theta = theta_rad(k);
            I_load_prime = I_load_prime_mag * exp(-1j * theta);
            V_drop = I_load_prime * (Req + 1j * Xeq);
            V1 = V_HV + V_drop;
            V1_mag = abs(V1);

            P_out = V_LV * I2 * cos(theta);
            copper_loss = (I_load_prime_mag^2) * Req;
            core_loss = (V1_mag^2) / Rc;
            P_in = P_out + copper_loss + core_loss;

            efficiency = P_out / P_in * 100;
            efficiencies(k) = efficiency;
        end

        plot(theta_deg, efficiencies, 'Color', colors(i,:), ...
             'DisplayName', sprintf('Load Factor = %.2f', lf), 'LineWidth', 1.5);
    end

    xlabel('Power Factor Angle (degrees)');
    ylabel('Efficiency (%)');
    title('b) Efficiency vs Power Factor Angle');
    legend('Location', 'best');
    grid on;
    hold off;

end
```

8.2 Simulation Results

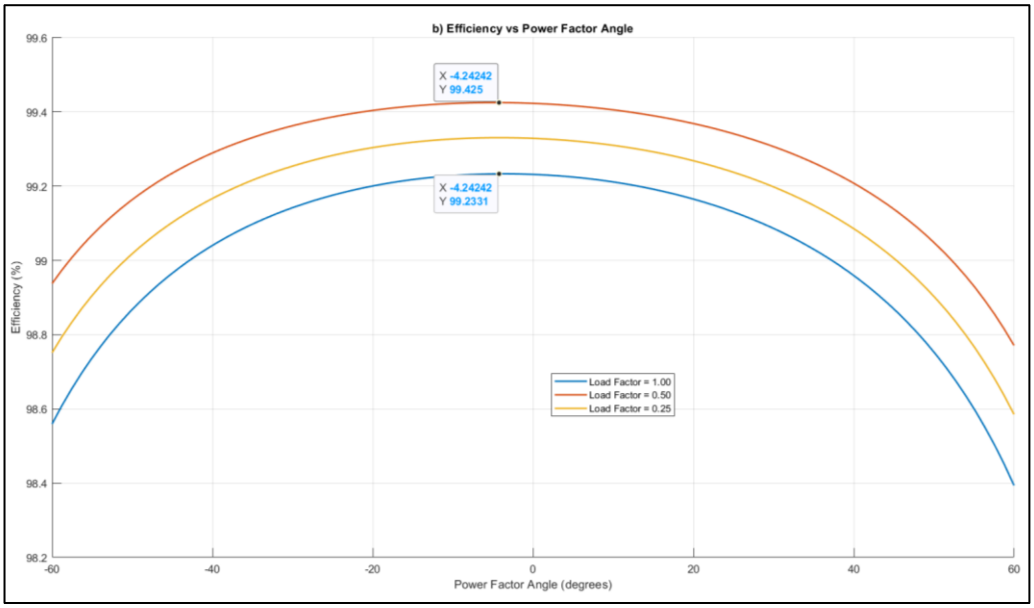

Figure 3 Efficiency vs Power Factor @Different Loads

8.3 Results Comment

The transformer efficiency was evaluated as a function of power factor angle across three different loading conditions: full load, half load, and quarter load. The results are plotted in **Figure 2** and reveal distinct trends that align with theoretical expectations.

From the simulation and plotted data, the following insights are observed:

- **Maximum efficiency occurs at unity power factor** ($\theta = 0°$). At this point, the load is purely resistive, minimizing reactive power and associated voltage drops across reactance components. As a result, both copper and core losses are balanced optimally.
- As the power factor angle shifts **towards lagging ($+\theta$)** or **leading ($-\theta$)**, the efficiency decreases. This is due to increased reactive components in the current, which either cause:
  - An **increase in copper losses** (when voltage drops below nominal),
  - Or an **increase in core losses** (when voltage rises above nominal).
- In terms of loading levels:
  - **Half load (orange)** shows the highest efficiency.
  - **Quarter load (green)** comes next.
  - **Full load (blue)** exhibits the lowest efficiency.

This order may seem counterintuitive, but it can be explained through the **maximum efficiency condition**:

- At full load, copper losses dominate due to high current, reducing efficiency despite delivering more power. On the other hand, quarter load has minimal copper losses, but fixed core losses dominate the denominator, resulting in a lower efficiency compared to half load. **Half load** strikes a better balance between these two losses, hence achieving **peak efficiency**.

9 Voltage Regulation vs Power Factor

9.1 Code snippet

```
function plot_regulation_vs_pf(I2_full, a, Req, Xeq, V_HV, load_factors, theta_deg)


    theta_rad = deg2rad(theta_deg);

    % Plot setup
    figure;
    hold on;
    colors = lines(length(load_factors));  % dynamic color generation

    for i = 1:length(load_factors)
        lf = load_factors(i);
        I2 = I2_full * lf;
        I_load_prime_mag = I2 / a;
        regulations = zeros(size(theta_rad));

        for k = 1:length(theta_rad)
            theta = theta_rad(k);
            I_load_prime = I_load_prime_mag * exp(-1j * theta);
            V_drop = I_load_prime * (Req + 1j * Xeq);
            V1 = V_HV + V_drop;
            V1_mag = abs(V1);

            regulation = (V1_mag - V_HV) / V_HV * 100;
            regulations(k) = regulation;
        end

        plot(theta_deg, regulations, 'Color', colors(i,:), ...
             'DisplayName', sprintf('Load Factor = %.2f', lf), 'LineWidth', 1.5);
    end

    xlabel('Power Factor Angle (degrees)');
    ylabel('Voltage Regulation (%)');
    title('c) Voltage Regulation vs Power Factor Angle');
    legend('Location', 'best');
    grid on;
    hold off;

end
```

9.2 Simulation Results

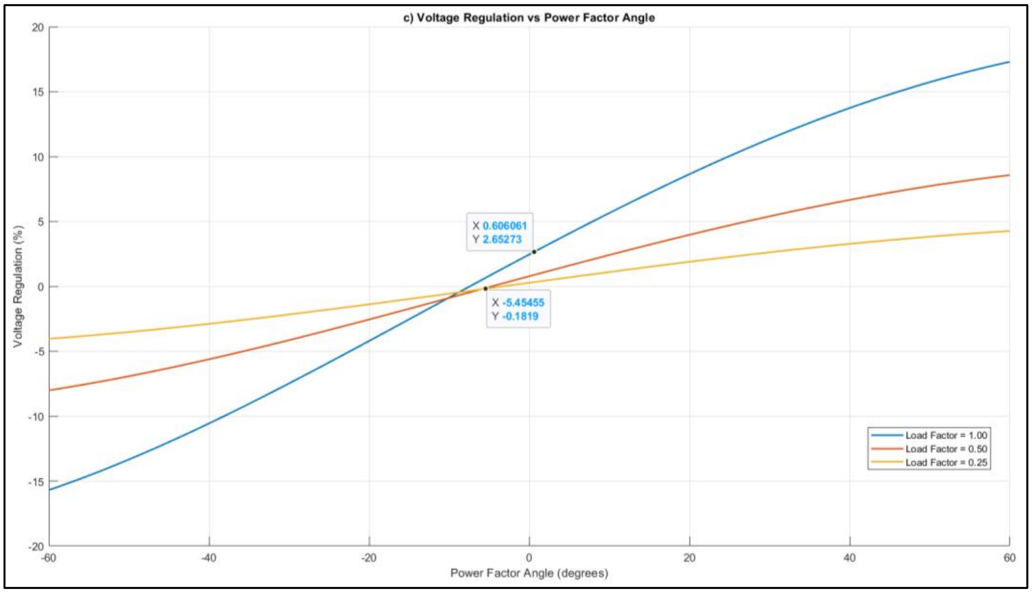

Figure 4 Voltage Regulation vs Power Factor @Different Loads

9.3 Results Comment

The variation in voltage regulation as a function of power factor angle across different loading conditions is shown in **Figure X: Voltage Regulation vs Power Factor Angle**. Voltage regulation (VR) is calculated using the formula:

$$VR=\frac{|V1|-|V1nom|}{|V1nom|}*100\%$$

Voltage regulation is minimized (improved) under light loading and leading power factor conditions.

The quarter load condition provides the most stable voltage regulation performance, making it ideal for applications requiring tight voltage tolerance.

10 Tapping

10.1 Code snippet

```
function check_tapping_range(I2_full, a, Req, Xeq, V_HV, load_factors, theta_deg)
% check_tapping_range - Verifies if HV voltage stays within ±5% tapping range
%                        for different load factors and power factor angles.
%   theta_deg    - Array of power factor angles in degrees (e.g., linspace(-54, 54, 100))

    theta_rad = deg2rad(theta_deg);
    tapping_upper = V_HV * 1.05;
    tapping_lower = V_HV * 0.95;

    % Set up plot
    figure;
    hold on;
    colors = lines(length(load_factors));  % Generate distinct colors
    labels = {'Full Load', 'Half Load', 'Quarter Load'};

    for i = 1:length(load_factors)
        lf = load_factors(i);
        I2 = I2_full * lf;
        V1_values = zeros(size(theta_rad));

        for k = 1:length(theta_rad)
            theta = theta_rad(k);
            I_load_prime = (I2 / a) * exp(-1j * theta);
            V_drop = I_load_prime * (Req + 1j * Xeq);
            V1 = V_HV + V_drop;
            V1_values(k) = abs(V1);
        end

        % Plot V1 over theta
        plot(theta_deg, V1_values, 'Color', colors(i,:), 'DisplayName', labels{i}, 'LineWidth', 1.5);

        % Find valid indices
        idx_valid = find(V1_values <= tapping_upper & V1_values >= tapping_lower);
        fprintf('Load factor: %.2f\n', lf);
        if ~isempty(idx_valid)
            min_valid_angle = theta_deg(min(idx_valid));
            max_valid_angle = theta_deg(max(idx_valid));
            fprintf(' ? HV within ±5%% tapping from %.1f° to %.1f°\n\n', min_valid_angle,
max_valid_angle);
        else
            fprintf(' ? HV voltage exceeds ±5%% tapping for all power factor angles.\n\n');
        end
    end

    % Add tapping limits to plot
    yline(tapping_upper, '--r', 'Upper Tapping Limit', 'LabelHorizontalAlignment', 'left', ...
          'LabelVerticalAlignment', 'bottom');
    yline(tapping_lower, '--b', 'Lower Tapping Limit', 'LabelHorizontalAlignment', 'left', ...
          'LabelVerticalAlignment', 'top');

    xlabel('Power Factor Angle (degrees)');
    ylabel('HV Terminal Voltage (V)');
    title('d) HV Voltage vs Power Factor Angle at Full, Half, and Quarter Load');
    legend('Location', 'best');
    grid on;
    hold off;

end
```

10.2 Simulation Results

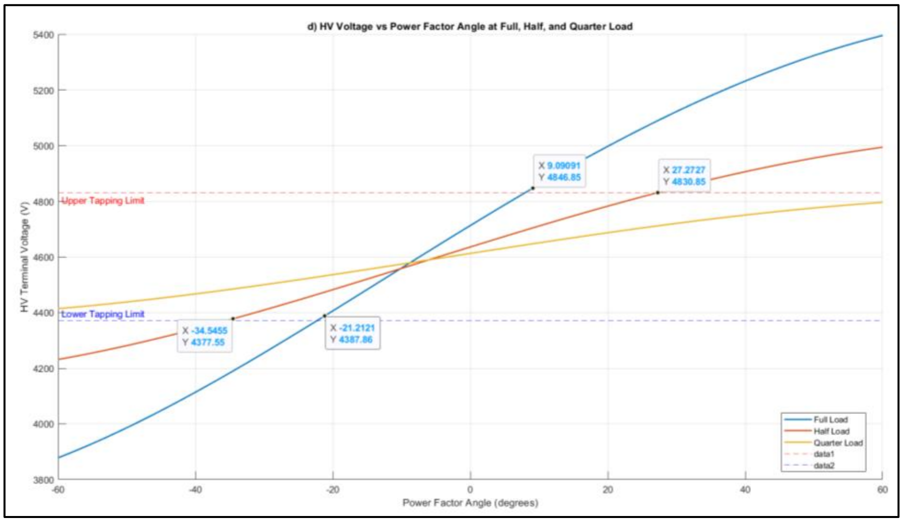

Figure 5 HV Voltage vs PF Angle with Tapping Limits

Load factor: 1.00
☑ HV within ±5% tapping from -21.2° to 7.9°

Load factor: 0.50
☑ HV within ±5% tapping from -34.5° to 26.1°

Load factor: 0.25
☑ HV within ±5% tapping from -60.0° to 60.0°

--- Tapping Note ---
The HV terminal voltage of the transformer changes with power factor.
The ±5% tapping range on the HV side equates to ±230 V (from 4370 V to 4830 V).
Refer to the plot to verify whether the voltage remains within these limits.

10.3 Results Comment

The transformer's **tap changing range**, typically set at ±5% of the nominal high-voltage (HV) value, is meant to regulate the output voltage under varying load and power factor conditions. However, the simulation results reveal critical limitations of this ±5% tap range under certain load scenarios.

- The permissible **HV tapping range** is from **4370 V to 4830 V**, based on the rated HV of **4600 V**.
- The actual required HV to maintain a constant LV of 240 V under **full load** varies significantly, from **3950.1 V** to **5348.2 V**, which **exceeds the ±5% range**.
- At **full load**, the HV only stays within the tapping range when the power factor angle is between **-22.4° and 7.1°**.
- At **half load**, the HV stays within the tapping range over a wider power factor range: **-35.5° to 26.7°**.
- At **quarter load**, **the entire power factor angle range** falls within the ±5% HV tapping window, meaning **no adjustment beyond the default tap range is needed**.

11 Appendix

```
clc;
clear;
close all;

% Transformer Ratings
S_rated = 75000;        % VA
V_HV = 4600;            % V
V_LV = 240;             % V
a = V_HV / V_LV;        % Turns ratio
% Full load secondary current
I2_full = S_rated / V_LV;

% Transformer impedances (in ohms)
R1 = 0.846;
R2 = 0.00261;
X1 = 26.8;
X2 = 0.0745;
Rc = 220000;
Xm = 112000;


% Total equivalent impedance on HV side
[Req, Xeq, Zeq] = refer_secondary_to_HV(R2, X2, a, R1, X1);

%Part a) HV Voltage vs Power Factor
% power factor varies from 0.6 pf leading through unity pf to 0.6 pf lagging
% add some saftey factor
theta_deg =linspace(-60, 60, 100);
theta_rad = deg2rad(theta_deg);
pf = cos(theta_deg);

% plot High Voltage vs Power Factor
plot_HV_voltage_vs_power_factor(I2_full, a, Req, Xeq, V_HV, theta_deg);


% Part b: Efficiency and regulation for different load factors
load_factors = [1.0, 0.5, 0.25];

% plot efficiency vs Power Factor
plot_efficiency_vs_pf(I2_full, a, Req, Xeq, Rc, V_HV, V_LV, load_factors, theta_deg);

% plot regulation vs Power Factor
plot_regulation_vs_pf(I2_full, a, Req, Xeq, V_HV, load_factors, theta_deg);

% draw tapping
check_tapping_range(I2_full, a, Req, Xeq, V_HV, load_factors, theta_deg)

% Note on ±5% voltage tapping range
fprintf('\n--- Tapping Note ---\n');
fprintf('The HV terminal voltage of the transformer changes with power factor.\n');
fprintf('The ±5%% tapping range on the HV side equates to ±230 V (from 4370 V to 4830 V).\n');
fprintf('Refer to the plot to verify whether the voltage remains within these limits.\n');


%************* Functions ********%
function [Req, Xeq, Zeq] = refer_secondary_to_HV(R2, X2, a, R1, X1)
% refer_secondary_to_HV - Refer secondary side impedance to the HV side
%
% Inputs:
%   R2 - Secondary side resistance (Ohms)
%   X2 - Secondary side reactance (Ohms)
%   a  - Turns ratio (V1/V2)
%   R1 - Primary side resistance (Ohms)
%   X1 - Primary side reactance (Ohms)
%
% Outputs:
%   Req - Total equivalent resistance on HV side (Ohms)
%   Xeq - Total equivalent reactance on HV side (Ohms)
```

```
%   Zeq - Total equivalent impedance on HV side (complex Ohms)

    % Refer secondary parameters to HV side
    R2_HV = R2 * a^2;
    X2_HV = X2 * a^2;

    % Total equivalent impedance on HV side
    Req = R1 + R2_HV;
    Xeq = X1 + X2_HV;
    Zeq = Req + 1i * Xeq;
end

function plot_HV_voltage_vs_power_factor(I2_full, a, Req, Xeq, V_HV, theta_deg)
% plot_HV_voltage_vs_power_factor - Plots the HV terminal voltage vs. power factor angle at full
load
%
% Inputs:
%   I2_full - Full-load secondary current (A)
%   a       - Turns ratio (V1/V2)
%   Req     - Equivalent resistance on HV side (Ohms)
%   Xeq     - Equivalent reactance on HV side (Ohms)
%   V_HV    - Nominal high-voltage terminal (V)
%   theta_deg - power factor angle range
%
% This function calculates and plots the HV terminal voltage for a range of power factor angles
% from 0.6 leading (-54°) to 0.6 lagging (+54°).

    % Define power factor angle range (degrees and radians)
    theta_rad = deg2rad(theta_deg);

    % Preallocate array for V1 values
    V1_values = zeros(size(theta_rad));

    % Compute HV terminal voltage for each angle
    for k = 1:length(theta_rad)
        theta = theta_rad(k);
        I_load_prime = (I2_full / a) * exp(-1j * theta);
        V_drop = I_load_prime * (Req + 1j * Xeq);
        V1 = V_HV + V_drop;
        V1_values(k) = abs(V1);
    end

    % Plotting
    figure;
    plot(theta_deg, V1_values, 'b', 'LineWidth', 1.5);
    xlabel('Power Factor Angle (degrees)');
    ylabel('HV Terminal Voltage (V)');
    title('a) HV Voltage vs Power Factor Angle at Full Load');
    grid on;

end

function plot_efficiency_vs_pf(I2_full, a, Req, Xeq, Rc, V_HV, V_LV, load_factors, theta_deg)
% plot_efficiency_vs_pf - Plots transformer efficiency vs power factor angle
%
% Inputs:
%   I2_full      - Full-load secondary current (A)
%   a            - Turns ratio (V1/V2)
%   Req          - Equivalent resistance on HV side (Ohms)
%   Xeq          - Equivalent reactance on HV side (Ohms)
%   Rc           - Core loss resistance referred to HV side (Ohms)
%   V_HV         - HV terminal voltage (V)
%   V_LV         - LV terminal voltage (V)
%   load_factors - Array of load factors (e.g., [1.0, 0.5, 0.25])
%   theta_deg    - Array of power factor angles in degrees (e.g., linspace(-54, 54, 100))

    theta_rad = deg2rad(theta_deg);
    pf = cos(theta_rad);  % Not used directly, but kept for interpretation

    % Plot setup
    figure;
```

```
    hold on;
    colors = lines(length(load_factors));  % dynamic color generation

    for i = 1:length(load_factors)
        lf = load_factors(i);
        I2 = I2_full * lf;
        I_load_prime_mag = I2 / a;
        efficiencies = zeros(size(theta_rad));

        for k = 1:length(theta_rad)
            theta = theta_rad(k);
            I_load_prime = I_load_prime_mag * exp(-1j * theta);
            V_drop = I_load_prime * (Req + 1j * Xeq);
            V1 = V_HV + V_drop;
            V1_mag = abs(V1);

            P_out = V_LV * I2 * cos(theta);
            copper_loss = (I_load_prime_mag^2) * Req;
            core_loss = (V1_mag^2) / Rc;
            P_in = P_out + copper_loss + core_loss;

            efficiency = P_out / P_in * 100;
            efficiencies(k) = efficiency;
        end

        plot(theta_deg, efficiencies, 'Color', colors(i,:), ...
             'DisplayName', sprintf('Load Factor = %.2f', lf), 'LineWidth', 1.5);
    end

    xlabel('Power Factor Angle (degrees)');
    ylabel('Efficiency (%)');
    title('b) Efficiency vs Power Factor Angle');
    legend('Location', 'best');
    grid on;
    hold off;

end

function plot_regulation_vs_pf(I2_full, a, Req, Xeq, V_HV, load_factors, theta_deg)
% plot_regulation_vs_pf - Plots transformer voltage regulation vs power factor angle
%
% Inputs:
%   I2_full      - Full-load secondary current (A)
%   a            - Turns ratio (V1/V2)
%   Req          - Equivalent resistance on HV side (Ohms)
%   Xeq          - Equivalent reactance on HV side (Ohms)
%   V_HV         - HV terminal voltage (V)
%   load_factors - Array of load factors (e.g., [1.0, 0.5, 0.25])
%   theta_deg    - Array of power factor angles in degrees (e.g., linspace(-54, 54, 100))

    theta_rad = deg2rad(theta_deg);

    % Plot setup
    figure;
    hold on;
    colors = lines(length(load_factors));  % dynamic color generation

    for i = 1:length(load_factors)
        lf = load_factors(i);
        I2 = I2_full * lf;
        I_load_prime_mag = I2 / a;
        regulations = zeros(size(theta_rad));

        for k = 1:length(theta_rad)
            theta = theta_rad(k);
            I_load_prime = I_load_prime_mag * exp(-1j * theta);
            V_drop = I_load_prime * (Req + 1j * Xeq);
            V1 = V_HV + V_drop;
            V1_mag = abs(V1);

            regulation = (V1_mag - V_HV) / V_HV * 100;
```

```
            regulations(k) = regulation;
        end

        plot(theta_deg, regulations, 'Color', colors(i,:), ...
             'DisplayName', sprintf('Load Factor = %.2f', lf), 'LineWidth', 1.5);
    end

    xlabel('Power Factor Angle (degrees)');
    ylabel('Voltage Regulation (%)');
    title('c) Voltage Regulation vs Power Factor Angle');
    legend('Location', 'best');
    grid on;
    hold off;

end

function check_tapping_range(I2_full, a, Req, Xeq, V_HV, load_factors, theta_deg)
% check_tapping_range - Verifies if HV voltage stays within ±5% tapping range
%                       for different load factors and power factor angles.
%
% Inputs:
%   I2_full      - Full-load secondary current (A)
%   a            - Turns ratio (V1/V2)
%   Req          - Equivalent resistance on HV side (Ohms)
%   Xeq          - Equivalent reactance on HV side (Ohms)
%   V_HV         - HV terminal voltage (V)
%   load_factors - Array of load factors (e.g., [1.0, 0.5, 0.25])
%   theta_deg    - Array of power factor angles in degrees (e.g., linspace(-54, 54, 100))

    theta_rad = deg2rad(theta_deg);
    tapping_upper = V_HV * 1.05;
    tapping_lower = V_HV * 0.95;

    % Set up plot
    figure;
    hold on;
    colors = lines(length(load_factors));  % Generate distinct colors
    labels = {'Full Load', 'Half Load', 'Quarter Load'};

    for i = 1:length(load_factors)
        lf = load_factors(i);
        I2 = I2_full * lf;
        V1_values = zeros(size(theta_rad));

        for k = 1:length(theta_rad)
            theta = theta_rad(k);
            I_load_prime = (I2 / a) * exp(-1j * theta);
            V_drop = I_load_prime * (Req + 1j * Xeq);
            V1 = V_HV + V_drop;
            V1_values(k) = abs(V1);
        end

        % Plot V1 over theta
        plot(theta_deg, V1_values, 'Color', colors(i,:), 'DisplayName', labels{i}, 'LineWidth',
1.5);

        % Find valid indices
        idx_valid = find(V1_values <= tapping_upper & V1_values >= tapping_lower);
        fprintf('Load factor: %.2f\n', lf);
        if ~isempty(idx_valid)
            min_valid_angle = theta_deg(min(idx_valid));
            max_valid_angle = theta_deg(max(idx_valid));
            fprintf(' ? HV within ±5%% tapping from %.1f° to %.1f°\n\n', min_valid_angle,
max_valid_angle);
        else
            fprintf(' ? HV voltage exceeds ±5%% tapping for all power factor angles.\n\n');
        end
    end

    % Add tapping limits to plot
    yline(tapping_upper, '--r', 'Upper Tapping Limit', 'LabelHorizontalAlignment', 'left', ...
```

```
            'LabelVerticalAlignment', 'bottom');
      yline(tapping_lower, '--b', 'Lower Tapping Limit', 'LabelHorizontalAlignment', 'left', ...
            'LabelVerticalAlignment', 'top');

      xlabel('Power Factor Angle (degrees)');
      ylabel('HV Terminal Voltage (V)');
      title('d) HV Voltage vs Power Factor Angle at Full, Half, and Quarter Load');
      legend('Location', 'best');
      grid on;
      hold off;

end
```