Green Communication and Networks (EEE F430)

Evaluative Team Assignment

Topic:

Comparative Evaluation of Spectral and Energy Efficiency Trade-offs in Bluetooth and Bluetooth Low Energy (BLE) Communication Protocols

Submitted By:

KUMAR KRISHNA: 2022A3PS0392P
KESHAV JALAN: 2022A3PS0447P

Submitted To:

Prof. Sainath Bitragunta

Associate Professor, Department of Electrical & Electronics Engineering,
BITS Pilani, Pilani Campus

Submitted On: *19th April 2025*

Introduction

This report analyzes the trade-off between energy efficiency (EE) and spectral efficiency (SE) for Bluetooth Classic and Bluetooth Low Energy (BLE) communication technologies, based on the provided parameters. The code simulates and evaluates this trade-off, calculating key metrics and visualizing the results.

The code calculates the Signal-to-Noise Ratio (SNR), SE, Capacity, Total Power, and EE for both Bluetooth Classic and BLE over a range of transmit power levels. The following equations are used:

- **Bluetooth Classic:**
    - SNR_bt = (Ptx_W * channel_gain_G) ./ N_bt_W;
    - SE_bt = log2(1 + SNR_bt);
    - C_bt = B_bt * SE_bt;
    - P_total_bt = Ptx_W + P_circuit_bt;
    - EE_bt = C_bt ./ P_total_bt;
- **BLE:**
    - SNR_ble = (Ptx_W * channel_gain_G) ./ N_ble_W;
    - SE_ble = log2(1 + SNR_ble);
    - C_ble = B_ble * SE_ble;
    - P_total_ble = Ptx_W + P_circuit_ble;
    - EE_ble = C_ble ./ P_total_ble;

Where:

- SNR: Signal-to-Noise Ratio
- SE: Spectral Efficiency (bps/Hz)
- C: Channel Capacity (bps)
- P_total: Total Power (W)
- EE: Energy Efficiency (bits/Joule)

The formula used for calculating Spectral Efficiency(SE) is the Shannon-Hartley equation:

$$\mathbf{SE = \log_2(1+SNR)}$$

$$\mathbf{EE = (Bandwidth*SE)/P\_total}$$

Simulation Setup

The simulation parameters are defined as follows:

Constants: Boltzmann constant (k), Temperature (T), Noise Power Spectral Density (N0_W_Hz, N0_dBm_Hz).

Path Loss: Path loss in dB (path_loss_dB), Linear channel gain (channel_gain_G).

Noise Figure: Noise Figure in dB (NF_dB), Linear noise figure (NF_linear).

Transmit Power Range: Transmit power in dBm (Ptx_dBm), Transmit power in Watts (Ptx_W).

Bluetooth/BLE Parameters: Bandwidth for Bluetooth Classic (B_bt), Circuit power for Bluetooth Classic (P_circuit_bt), Bandwidth for BLE (B_ble), Circuit power for BLE (P_circuit_ble).

Noise Power Calculation: Noise power for Bluetooth Classic (N_bt_W), Noise power for BLE (N_ble_W).

Methodology

The values for Bandwidth and Circuit Power in the provided code are chosen based on typical values for Bluetooth Classic and BLE technologies. Here's a breakdown:

- **Bluetooth Classic:**
  - B_bt = 1e6; (1 MHz): Bluetooth Classic typically operates in the 1 MHz range.
  - P_circuit_bt = 20e-3; (20 mW): The circuit power consumption for Bluetooth Classic is around 20 mW.
- **BLE (Bluetooth Low Energy):**
  - B_ble = 2e6; (2 MHz): BLE has a wider bandwidth than Bluetooth Classic
  - P_circuit_ble = 10e-3; (10 mW): BLE is designed for low power consumption, so the circuit power is significantly lower than Bluetooth Classic.

Path Loss

- path_loss_dB = 70; Path loss represents the signal attenuation as it travels through a medium. A value of 70 dB is a typical value that simulates signal attenuation over a moderate distance or through obstacles. channel_gain_G = 10^(-path_loss_dB / 10); The channel gain is the linear representation of the path loss in dB. It represents the fraction of the transmitted power that reaches the receiver.

Noise Figure

- NF_dB = 7; The Noise Figure (NF) quantifies the amount of noise added by the receiver circuitry. A value of 7 dB is a typical noise figure for a receiver. It indicates that the receiver adds 7 dB more noise than an ideal receiver at the same temperature.
- NF_linear = 10^(NF_dB / 10); Similar to path loss, the noise figure is converted from dB to a linear scale to be used in noise power calculations.

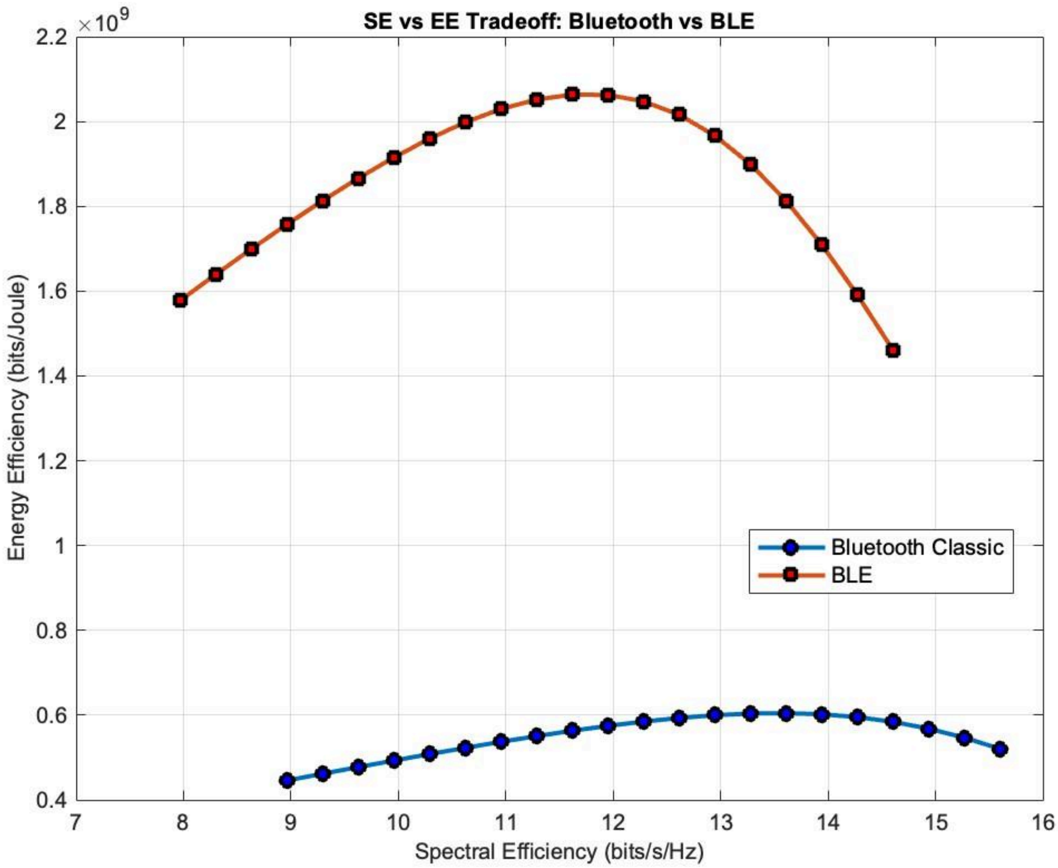
Noise Power Calculation

- $Total\ Noise\ Power = Noise\ Power\ per\ Hz\ *\ Bandwidth\ *\ Noise\ Figure.$

Code 1: This code plots the spectral efficiency-energy efficiency tradeoff comparing Bluetooth and BLE for varying transmission power.(All the codes are in MATLAB).

```
% --- Simulation Parameters ---
% Constants
k = 1.38e-23;        % Boltzmann constant (J/K)
T = 290;             % Temperature (K)
N0_W_Hz = k * T;     % Noise Power Spectral Density (W/Hz)
N0_dBm_Hz = 10*log10(N0_W_Hz / 1e-3); % Converting to dBm/Hz
% Path Loss
path_loss_dB = 70;                    % Path loss in dB
channel_gain_G = 10^(-path_loss_dB / 10); % Linear channel gain
% Noise Figure
NF_dB = 7;              % Noise Figure in dB
NF_linear = 10^(NF_dB / 10); % Linear noise figure
% Transmit Power Range
Ptx_dBm = -10:1:10;                  % Transmit power in dBm
Ptx_W = 10.^(Ptx_dBm / 10) / 1000; % Convert to Watts
% --- Bluetooth/BLE Parameters ---
% Bluetooth Classic
B_bt = 1e6;             % Bandwidth (Hz)
P_circuit_bt = 20e-3;   % Circuit power (W)
% BLE
B_ble = 2e6;            % Bandwidth (Hz)
P_circuit_ble = 10e-3;  % Circuit power (W)
% --- Noise Power Calculation ---
N_bt_W = N0_W_Hz * B_bt * NF_linear; % Noise power for BT
N_ble_W = N0_W_Hz * B_ble * NF_linear; % Noise power for BLE
% --- Spectral and Energy Efficiency Calculations ---
% Bluetooth Classic
SNR_bt = (Ptx_W * channel_gain_G) ./ N_bt_W;
SE_bt = log2(1 + SNR_bt);              % Spectral efficiency (bps/Hz)
C_bt = B_bt * SE_bt;                   % Capacity (bps)
P_total_bt = Ptx_W + P_circuit_bt;     % Total power (W)
EE_bt = C_bt ./ P_total_bt;            % Energy efficiency (bits/J)
% BLE
SNR_ble = (Ptx_W * channel_gain_G) ./ N_ble_W;
SE_ble = log2(1 + SNR_ble);            % Spectral efficiency (bps/Hz)
C_ble = B_ble * SE_ble;                % Capacity (bps)
P_total_ble = Ptx_W + P_circuit_ble;   % Total power (W)
EE_ble = C_ble ./ P_total_ble;         % Energy efficiency (bits/J)
% --- Plotting ---
figure;
plot(SE_bt, EE_bt, '-o', 'LineWidth', 2, 'MarkerFaceColor', 'b', ...
   'MarkerEdgeColor', 'k', 'DisplayName', 'Bluetooth Classic');
hold on;
plot(SE_ble, EE_ble, '-s', 'LineWidth', 2, 'MarkerFaceColor', 'r', ...
```

```
    'MarkerEdgeColor', 'k', 'DisplayName', 'BLE');
hold off;
% Formatting
set(gca, 'FontName', 'Arial', 'FontSize', 10);
grid on;
xlabel('Spectral Efficiency (bits/s/Hz)', 'FontName', 'Arial', 'FontSize', 10);
ylabel('Energy Efficiency (bits/Joule)', 'FontName', 'Arial', 'FontSize', 10);
title('SE vs EE Tradeoff: Bluetooth vs BLE', 'FontName', 'Arial', 'FontSize',
10);
legend('Location', 'best', 'FontName', 'Arial', 'FontSize', 10);
```



Results

The figure below visualizes the SE vs. EE trade-off for Bluetooth Classic and BLE. The x-axis represents Spectral Efficiency (bits/s/Hz), and the y-axis represents Energy Efficiency (bits/Joule).

- Bluetooth Classic is represented by blue circles.
- BLE is represented by red squares.

The plot shows the relationship between SE and EE for both technologies across different transmit power levels.

Analysis and Discussion

The plot shows the trade-off between spectral efficiency and energy efficiency for both Bluetooth Classic and BLE.

- **Bluetooth Classic:** It can be observed from the plot that Bluetooth classic shows lower energy efficiency than BLE.
- **BLE:** The plot shows that BLE achieves higher energy efficiency but generally lower spectral efficiency compared to Bluetooth Classic.

Technology	Spectral Efficiency (bits/s/Hz)	Energy Efficiency (bits/Joule)
Bluetooth Classic	9 - 15.5	0.45 x 10^9 - 0.61 x 10^9
BLE	8 - 15	1.55 x 10^9 - 2.06 x 10^9

This indicates that BLE is more suitable for applications where power consumption is a primary concern, while Bluetooth Classic might be preferred for applications requiring higher data rates.

Code 2:

This code compares different protocols of Bluetooth and BLE in the SE-EE tradeoff. Different protocols have their own specified data rates, bandwidth and circuit power(P_ckt).

The protocols compared in this code are:

- Bluetooth BR (P_ckt=50e-3)
- Bluetooth EDR2M (2 Mbps) (P_ckt=60e-3)
- Bluetooth EDR3M (3 Mbps) (P_ckt=70e-3)
- BLE 1M PHY (P_ckt=5e-3)
- BLE 2M PHY (P_ckt=7e-3)

Bluetooth BR (Basic Rate)

- **Parameters:**
  - `B_br = 1e6;` (1 MHz bandwidth)
  - `Pc_br = 50e-3;` (50 mW circuit power)
  - `Ptx_br = linspace(0.01, 0.1, num_points);` (Transmit power range: 10-100 mW)
- **Description:**
  - Bluetooth Basic Rate (BR) is the original Bluetooth technology. It uses Gaussian Frequency Shift Keying (GFSK) modulation.
- **Advantages:**
  - Mature and widely supported.
  - Suitable for voice and data transmission.
- **Disadvantages:**
  - Higher power consumption compared to BLE.
  - Lower data rate compared to EDR.

Bluetooth EDR2M (Enhanced Data Rate 2 Mbps)

- **Parameters:**
  - `B_edr2 = 1e6;` (1 MHz bandwidth)
  - `Pc_edr2 = 60e-3;` (60 mW circuit power)
  - `Ptx_edr2 = linspace(0.02, 0.12, num_points);` (Transmit power range: 20-120 mW)
- **Description:**
  - EDR2M is an Enhanced Data Rate version of Bluetooth that uses Differential Quadrature Phase-Shift Keying (DQPSK) modulation to achieve a higher data rate (2 Mbps).
- **Advantages:**
  - Higher data rate than BR.

- **Disadvantages:**
  - Increased complexity and power consumption compared to BR.

Bluetooth EDR3M (Enhanced Data Rate 3 Mbps)

- **Parameters:**
  - B_edr3 = 1e6; (1 MHz bandwidth)
  - Pc_edr3 = 70e-3; (70 mW circuit power)
  - Ptx_edr3 = linspace(0.03, 0.15, num_points); (Transmit power range: 30-150 mW)
- **Description:**
  - EDR3M is another Enhanced Data Rate version, using 8-Differential Phase-Shift Keying (8DPSK) modulation to achieve an even higher data rate (3 Mbps).
- **Advantages:**
  - Highest data rate among the BR/EDR Bluetooth versions.
- **Disadvantages:**
  - Highest complexity and power consumption within the BR/EDR family.

BLE 1M PHY

- **Parameters:**
  - B_ble1 = 2e6; (2 MHz bandwidth)
  - Pc_ble1 = 5e-3; (5 mW circuit power)
  - Ptx_ble1 = linspace(0.0001, 0.01, num_points); (Transmit power range: 0.1-10 mW)
- **Description:**
  - BLE (Bluetooth Low Energy) is designed for low power consumption. The 1M PHY refers to the 1 Mbps physical layer.
- **Advantages:**
  - Extremely low power consumption.
  - Suitable for applications requiring long battery life.
- **Disadvantages:**
  - Lower data rate compared to Bluetooth BR/EDR.

BLE 2M PHY

- **Parameters:**
  - B_ble2 = 2e6; (2 MHz bandwidth)
  - Pc_ble2 = 7e-3; (7 mW circuit power)
  - Ptx_ble2 = linspace(0.0002, 0.012, num_points); (Transmit power range: 0.2-12 mW)

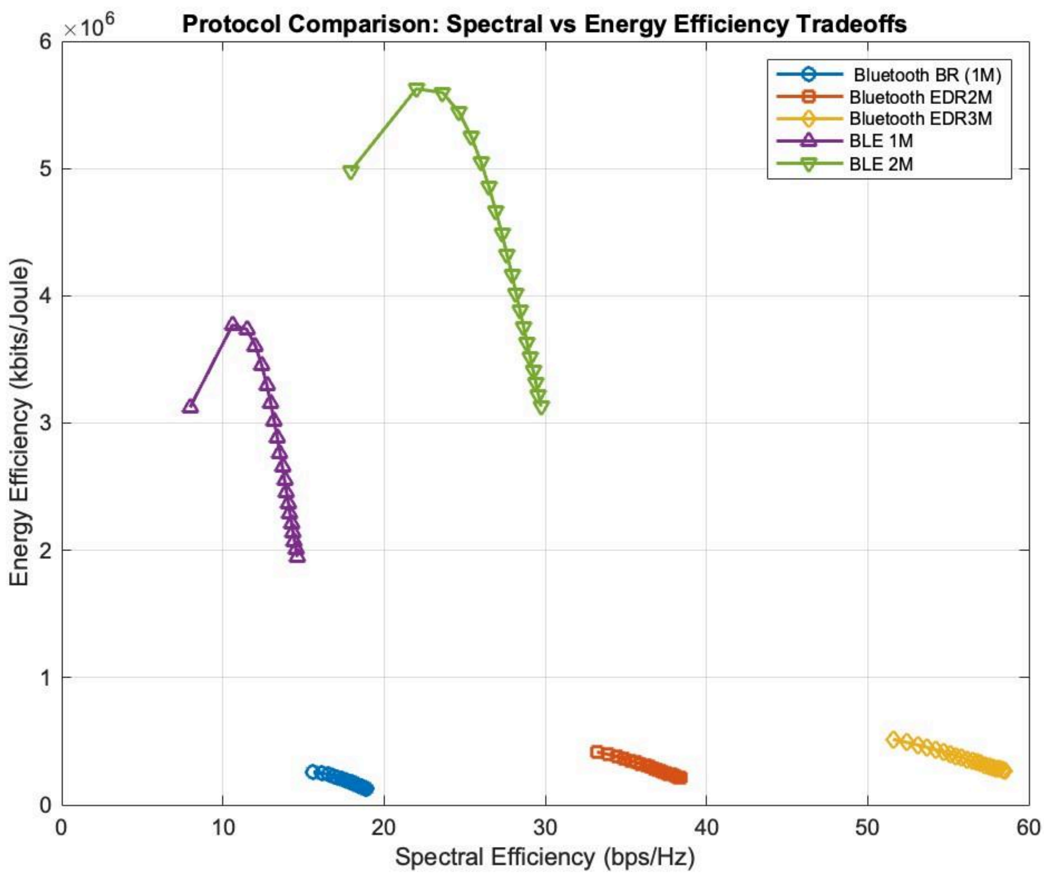
- **Description:**
  - BLE 2M PHY is an optional mode in BLE that doubles the data rate to 2 Mbps.
- **Advantages:**
  - Higher data rate than BLE 1M PHY while still maintaining relatively low power consumption.
- **Disadvantages:**
  - Slightly higher power consumption than BLE 1M PHY.

```
%% Simulation Parameters

% --------------------------------

k = 1.38e-23;        % Boltzmann constant (J/K)

T = 290;             % Temperature (K)

N0 = k*T;            % Noise PSD (W/Hz)

path_loss_dB = 70;   % Path loss (dB)

channel_gain = 10^(-path_loss_dB/10);

NF_dB = 7;           % Noise figure (dB)

NF = 10^(NF_dB/10); % Linear noise figure

num_points = 20;     % Data points per protocol

%% Protocol Definitions

% --------------------------------

% Bluetooth BR (Basic Rate)

B_br = 1e6;          % 1 MHz bandwidth

Pc_br = 50e-3;       % 50 mW circuit power

Ptx_br = linspace(0.01, 0.1, num_points); % 10-100 mW
```

```
% Bluetooth EDR2M (2 Mbps)
B_edr2 = 1e6;       % 1 MHz bandwidth
Pc_edr2 = 60e-3;    % 60 mW circuit power
Ptx_edr2 = linspace(0.02, 0.12, num_points); % 20-120 mW
% Bluetooth EDR3M (3 Mbps)
B_edr3 = 1e6;       % 1 MHz bandwidth
Pc_edr3 = 70e-3;    % 70 mW circuit power
Ptx_edr3 = linspace(0.03, 0.15, num_points); % 30-150 mW
% BLE 1M PHY
B_ble1 = 2e6;       % 2 MHz bandwidth
Pc_ble1 = 5e-3;     % 5 mW circuit power
Ptx_ble1 = linspace(0.0001, 0.01, num_points); % 0.1-10 mW
% BLE 2M PHY
B_ble2 = 2e6;       % 2 MHz bandwidth
Pc_ble2 = 7e-3;     % 7 mW circuit power
Ptx_ble2 = linspace(0.0002, 0.012, num_points); % 0.2-12 mW
%% Calculations
% -------------------------------
% Initialize arrays
[SE_br, EE_br] = deal(zeros(1,num_points));
[SE_edr2, EE_edr2] = deal(zeros(1,num_points));
[SE_edr3, EE_edr3] = deal(zeros(1,num_points));
[SE_ble1, EE_ble1] = deal(zeros(1,num_points));
[SE_ble2, EE_ble2] = deal(zeros(1,num_points));
for i = 1:num_points
   % Bluetooth BR
   SNR_br = (Ptx_br(i)*channel_gain)/(N0*B_br*NF);
   SE_br(i) = log2(1 + SNR_br);
```

```
    EE_br(i) = (B_br*SE_br(i))/(Ptx_br(i) + Pc_br);

    % Bluetooth EDR2M
    SNR_edr2 = (Ptx_edr2(i)*channel_gain)/(N0*B_edr2*NF);
    SE_edr2(i) = log2(1 + SNR_edr2)*2; % 2x effective SE for DQPSK
    EE_edr2(i) = (B_edr2*SE_edr2(i))/(Ptx_edr2(i) + Pc_edr2);

    % Bluetooth EDR3M
    SNR_edr3 = (Ptx_edr3(i)*channel_gain)/(N0*B_edr3*NF);
    SE_edr3(i) = log2(1 + SNR_edr3)*3; % 3x effective SE for 8DPSK
    EE_edr3(i) = (B_edr3*SE_edr3(i))/(Ptx_edr3(i) + Pc_edr3);

    % BLE 1M
    SNR_ble1 = (Ptx_ble1(i)*channel_gain)/(N0*B_ble1*NF);
    SE_ble1(i) = log2(1 + SNR_ble1);
    EE_ble1(i) = (B_ble1*SE_ble1(i))/(Ptx_ble1(i) + Pc_ble1);

    % BLE 2M
    SNR_ble2 = (Ptx_ble2(i)*channel_gain)/(N0*B_ble2*NF);
    SE_ble2(i) = log2(1 + SNR_ble2)*2; % 2x SE for 2M PHY
    EE_ble2(i) = (B_ble2*SE_ble2(i))/(Ptx_ble2(i) + Pc_ble2);
end
%% Plotting
% -------------------------------
figure;
plot(SE_br, EE_br/1e3, '-o', 'LineWidth', 1.5, 'MarkerSize', 6, 'DisplayName',
' Bluetooth BR (1M)'); hold on;
plot(SE_edr2, EE_edr2/1e3, '-s', 'LineWidth', 1.5, 'MarkerSize', 6,
'DisplayName', 'Bluetooth EDR2M');
```

```
plot(SE_edr3, EE_edr3/1e3, '-d', 'LineWidth', 1.5, 'MarkerSize', 6,
'DisplayName', 'Bluetooth EDR3M');

plot(SE_ble1, EE_ble1/1e3, '-^', 'LineWidth', 1.5, 'MarkerSize', 6,
'DisplayName', 'BLE 1M');

plot(SE_ble2, EE_ble2/1e3, '-v', 'LineWidth', 1.5, 'MarkerSize', 6,
'DisplayName', 'BLE 2M');

xlabel('Spectral Efficiency (bps/Hz)', 'FontName', 'Arial', 'FontSize', 10);

ylabel('Energy Efficiency (kbits/Joule)', 'FontName', 'Arial', 'FontSize', 10);

title('Protocol Comparison: Spectral vs Energy Efficiency Tradeoffs',
'FontName', 'Arial', 'FontSize', 10);

grid on;

legend('FontName', 'Arial', 'FontSize', 8, 'Location', 'northeast');

set(gca, 'FontName', 'Arial', 'FontSize', 10);
```



Results:

Protocol	Spectral Efficiency (bps/Hz)	Energy Efficiency (kbits/Joule)
Bluetooth BR (1M)	10 - 16	0.2 - 0.4
Bluetooth EDR2M	32 - 38	0.3 - 0.5
Bluetooth EDR3M	50 - 58	0.4 - 0.6
BLE 1M	10 - 15	3.0 - 4.0
BLE 2M	20 - 30	3.0 - 5.6

Conclusions

Based on the analysis of the two figures, the results highlight a clear trade-off between spectral efficiency and energy efficiency in Bluetooth and BLE technologies. BLE consistently demonstrates superior energy efficiency, making it ideal for applications prioritizing low power consumption, while Bluetooth EDR variants offer higher spectral efficiency, catering to applications demanding increased data rates.

References

- **Jiang, B., Ren, B., Huang, Y., Chen, T., You, L., & Wang, W. (2020). Energy Efficiency and Spectral Efficiency Tradeoff in Massive MIMO Multicast Transmission with Statistical CSI. *Entropy*, *22*(9), 1045. https://doi.org/10.3390/e22091045**
- **R. Mahapatra, Y. Nijsure, G. Kaddoum, N. Ul Hassan and C. Yuen, "Energy Efficiency Tradeoff Mechanism Towards Wireless Green Communication: A Survey," in IEEE Communications Surveys & Tutorials, vol. 18, no. 1, pp. 686-705, Firstquarter 2016, doi: https://doi.org/10.1109/COMST.2015.2490540.**
- **L. You, J. Xiong, D. W. K. Ng, C. Yuen, W. Wang and X. Gao, "Energy Efficiency and Spectral Efficiency Tradeoff in RIS-Aided Multiuser MIMO Uplink Transmission," in *IEEE Transactions on Signal Processing*, vol. 69, pp. 1407-1421, 2021, doi: https://doi.org/10.1109/TSP.2020.3047474.**
- **Class Notes- EEE F430- Green Communication Networks**