Table of Contents

Channel capacity over EsNo

Get the channel capacity as a function of the EsNo, for different modulation techniques (M-QAM, M-PAM, M-PSK)

```
clc; clear; close all;
```

Parameters

```
mod_type = "QAM";                   % Modulation type.
M = [4, 16, 64];                    % Number of symbols. Modulation order.
symbol_qtty = 1e4;                  % Amount of symbols used in the
 simulation.
EsNo_dB = -10:2:40;                 % Repeat the simulation for every value of
 EsNo.
PlosPnlos_dB = 20;                  % Ratio between LOS and NLOS power.
channel_type = ChannelTypes.AWGN;
```

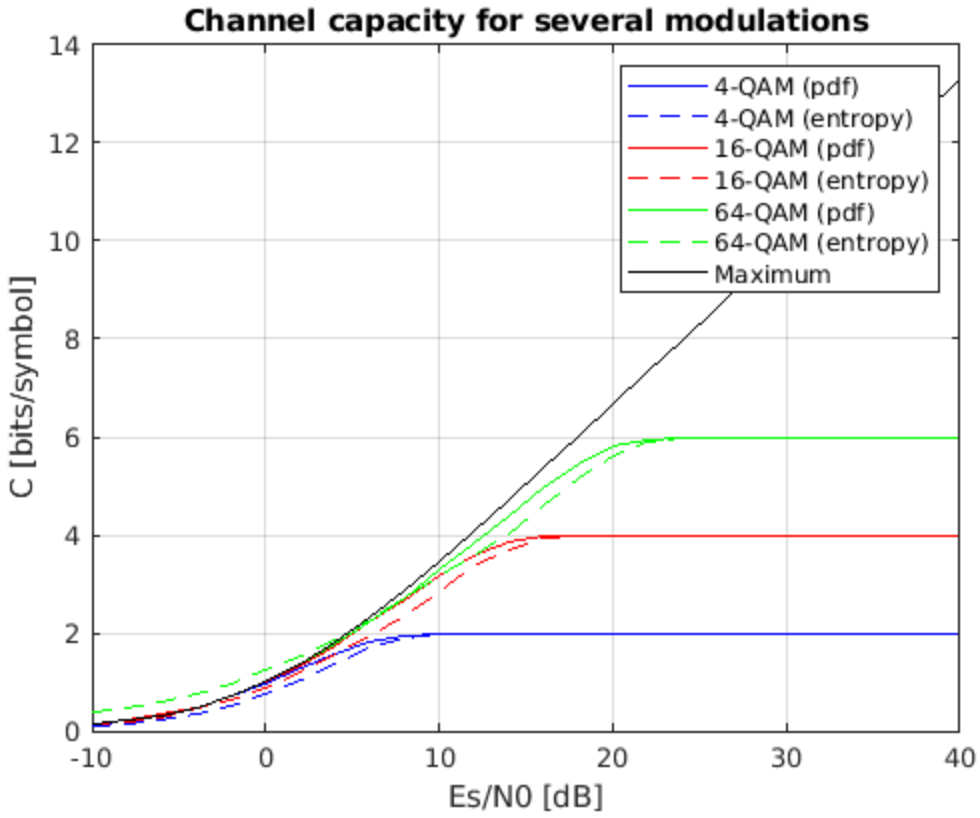
Intermidiate variables

```
colors = ["b", "r", "g", "c", "m", "k"];    % Colors for plotting
legendString = cell(1, 2*length(M)+1);      % For legend in plot, as "16-QAM"

% Channel capacity calculated with entropy formula
C_entropy = zeros(1, length(EsNo_dB));

% Channel capacity calculated with Gaussian probabilty density function
C_pdf = zeros(1, length(EsNo_dB));
```

Communication

```
for m=1:length(M)
    d = randi([0, M(m)-1], 1, symbol_qtty);     % Input symbol stream.
    d_r = zeros(1, symbol_qtty);                % Ouput symbol stream.

    % Modulator
    [s, constellation] = Modulator.modulate(d, mod_type, M(m));

    for i=1:length(EsNo_dB)
        % Channel
        switch channel_type
            case ChannelTypes.AWGN
                [r, h_c, N0] = Channel.add_awgn_noise(s, EsNo_dB(i), 1);
            case ChannelTypes.Rayleigh
```

```
                [r, h_c, N0] = Channel.add_rayleigh_noise(s, EsNo_dB(i), 1);
            case ChannelTypes.Ricean
                [r, h_c, N0] = Channel.add_ricean_noise(s, EsNo_dB(i), 1,
 PlosPnlos_dB);
        end

        % Demodulator
        r = Demodulator.flat_fading_equalizer(r, h_c);
        d_r = Demodulator.demodulate(r, mod_type, M(m), constellation);

        % Calculate channel capacity
        C_pdf(i) = Theory.channel_capacity_from_pdf(r, constellation, h_c,
 N0);
        C_entropy(i) = Scope.channel_capacity(M(m),d,d_r);
    end

    % Plot the channel capacity for the current modulation order M(m).
    plot(EsNo_dB, C_pdf, Color=colors(m), LineStyle='-'); hold on;
    plot(EsNo_dB, C_entropy, Color=colors(m), LineStyle="--"); hold on;

    legendString{2*m-1} = strcat(num2str(M(m)), "-", mod_type, " (pdf)");
    legendString{2*m}   = strcat(num2str(M(m)), "-", mod_type, " (entropy)");
end

% Calculate theoretical maximum channel capacity, with shannon formula
C_maximum = Theory.maximum_channel_capacity(EsNo_dB, h_c);
plot(EsNo_dB, C_maximum, Color=colors(end), LineStyle='-'); hold on;
legendString{end} = "Maximum";

legend(legendString);
xlabel("Es/N0 [dB]");
ylabel("C [bits/symbol]");
title("Channel capacity for several modulations");
grid on;
```



Published with MATLAB® R2022b