Table of Contents

Paper

```
% Title: Multi-RIS-aided Wireless Systems: Statistical
 Characterization and Performance Analysis
% Authors : Tri Nhu Do, Georges Kaddoum, Thanh Luan Nguyen, Daniel
 Benevides da Costa, Zygmunt J. Haas
% Online: https://github.com/trinhudo/Multi-RIS
% Version: 12-Sep-2021

% Energy efficiency of ERA and ORA scheme
```

Setting

```
clear all
close all

sim_trials = 1e5; % Number of simulation trails

N_RIS = 5; % Number of RISs

L = 25*ones(1, N_RIS); % Element setting L1

kappa_nl = 1; % Amplitude reflection coefficient

% Nakagami m parameter
m_0 = 2.5 + rand; % Scale parameter, Heuristic setting
m_h = 2.5 + rand(N_RIS, 1);
m_g = 2.5 + rand(N_RIS, 1);

% Network area
% ------------

x_area_min = 0;
x_area_max = 100;
y_area_min = 0;
y_area_max = 10;

% Source location
x_source = x_area_min;
y_source = y_area_min;

% Destination location
x_des = x_area_max;
y_des = y_area_min;
```

```
% Random topology
x_RIS = x_area_min + (x_area_max-x_area_min)*rand(N_RIS, 1); %
 [num_RIS x 1] vector
y_RIS = y_area_min + (y_area_max-y_area_min)*rand(N_RIS, 1);

% Compute location of nodes
pos_source = [x_source, y_source];
pos_des = [x_des, y_des];
pos_RIS = [x_RIS, y_RIS]; % [num_RIS x 2] matrix

% Compute distances
d_sr = sqrt(sum((pos_source - pos_RIS).^2 , 2)); % [num_RIS x 1]
 vector
d_rd = sqrt(sum((pos_RIS - pos_des).^2 , 2));
d_sd = sqrt(sum((pos_source - pos_des).^2 , 2));

% Path-loss model
% ---------------

% Carrier frequency (in GHz)
fc = 3; % GHz

% 3GPP Urban Micro in 3GPP TS 36.814
pathloss_NLOS = @(x) db2pow(-22.7 - 26*log10(fc) - 36.7*log10(x));

antenna_gain_S = db2pow(5); % Source antenna gain, dBi
antenna_gain_RIS = db2pow(5); % Gain of each element of a RIS, dBi
antenna_gain_D = db2pow(0); % Destination antenna gain, dBi

% Noise power and Transmit power P_S
% ----------------------------------

% Bandwidth
BW = 10e6; % 10 MHz

% Noise figure (in dB)
noiseFiguredB = 10;

% Compute the noise power in dBm
sigma2dBm = -174 + 10*log10(BW) + noiseFiguredB; % -94 dBm
sigma2 = db2pow(sigma2dBm);

P_S_dB = -30:.1:50; % Transmit power of the source, dBm, e.g., 200mW =
 23dBm

SNR_dB = P_S_dB - sigma2dBm; % Average transmit SNR, dB = dBm - dBm
```

Simplified Simulation

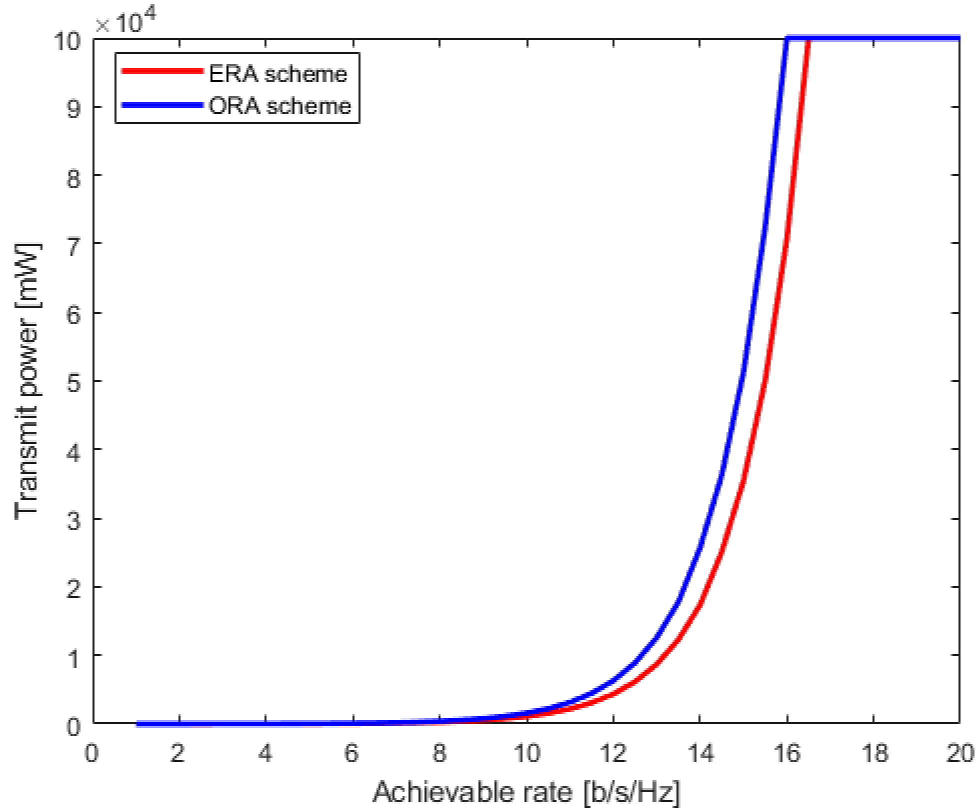
```
% Direct channel h_0
Omg_0 = pathloss_NLOS(d_sd)*antenna_gain_S; % Omega of S->D link
```

```
h_0 = random('Naka', m_0, Omg_0, [1, sim_trials]);
SNR_h0 = abs(h_0).^2;

V_n = zeros(N_RIS, sim_trials);
Omg_h = zeros(N_RIS, 1);
Omg_g = zeros(N_RIS, 1);

for nn = 1:N_RIS
    for kk = 1:L(nn)
        Omg_h(nn) =
 pathloss_NLOS(d_sr(nn))*antenna_gain_S*antenna_gain_RIS*L(nn); %
 Omega S->R
        Omg_g(nn) =
 pathloss_NLOS(d_rd(nn))*antenna_gain_RIS*L(nn)*antenna_gain_D; %
 Omega R->D

        h_nl = random('Naka', m_h(nn), Omg_h(nn), [1, sim_trials]);
        g_nl = random('Naka', m_g(nn), Omg_g(nn), [1, sim_trials]);

        U_nl = kappa_nl * h_nl .* g_nl;

        V_n(nn, :) = V_n(nn, :) + U_nl;
    end
end

% ERA scheme

T_ERA   = sum(V_n, 1);
Z_ERA   = h_0 + T_ERA; % Magnitude of the e2e channel
Z2_ERA  = Z_ERA.^2; % Squared magnitude of the e2e channel

% ORA scheme

[V_M_ORA, id_RIS] = max(V_n, [], 1); % V_M for the best RIS
R_ORA   = h_0 + V_M_ORA; % Magnitude of the e2e channel
R2_ORA  = R_ORA.^2; % Squared magnitude of the e2e channel

% Performance metrics

for isnr = 1:length(SNR_dB)
    %     fprintf('EC, SNR = % d \n', SNR_dB(isnr));
    snr = 10^(SNR_dB(isnr)/10);

    EC_ERA_sim(isnr) = mean(log2(1+snr*Z2_ERA));
    EC_ORA_sim(isnr) = mean(log2(1+snr*R2_ORA));

end
```

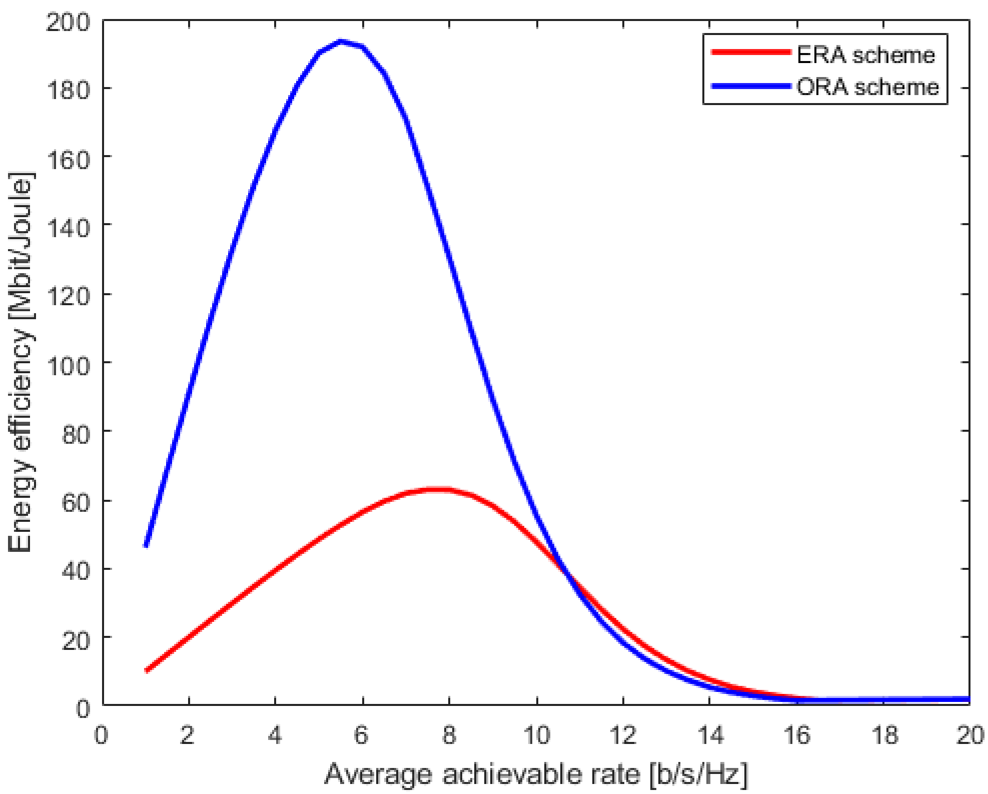
Numerical Results

```
% Transmit Power
% --------------
```

```
EC = 1:.5:20;
P_tx_ERA = zeros(1, length(EC));
P_tx_ORA = zeros(1, length(EC));

for ii = 1:length(EC)
    idx = find(EC_ERA_sim <= EC(ii));
    P_tx_ERA(ii) = db2pow(P_S_dB(idx(end))); %
 10^(SNR_dB(idx(end))/10);
    idx = find(EC_ORA_sim <= EC(ii));
    P_tx_ORA(ii) = db2pow(P_S_dB(idx(end))); %
 10^(SNR_dB(idx(end))/10);
end

figure;

plot(EC, P_tx_ERA, 'r-', 'linewidth', 2); hold on
plot(EC, P_tx_ORA, 'b-', 'linewidth', 2); hold on

xlabel('Achievable rate [b/s/Hz]')
ylabel('Transmit power [mW]')
legend('ERA scheme', 'ORA scheme', 'location', 'best')

% Engery Efficiency
% -----------------

P_c_S = db2pow(10); % Circuit dissipated power at S = 10 dBm
P_c_D = db2pow(10); % Circuit dissipated power at S = 10 dBm
P_c_element = 7.8; % mW

for kk=1:length(EC)
    P_total_ERA(kk) = P_tx_ERA(kk) + sum(L)*P_c_element + P_c_S +
 P_c_D;
    P_total_ORA(kk) = P_tx_ORA(kk) + max(L)*P_c_element + P_c_S +
 P_c_D;
end

EE_ERA = BW*EC./P_total_ERA/1e3;
EE_ORA = BW*EC./P_total_ORA/1e3;

figure;

plot(EC, EE_ERA, 'r-', 'linewidth', 2); hold on
plot(EC, EE_ORA, 'b-', 'linewidth', 2); hold on
xlabel('Average achievable rate [b/s/Hz]')
ylabel('Energy efficiency [Mbit/Joule]')
legend('ERA scheme', 'ORA scheme', ...
    'location', 'best')
```



×10^4
ERA scheme
ORA scheme
Transmit power [mW]
Achievable rate [b/s/Hz]



ERA scheme
ORA scheme
Energy efficiency [Mbit/Joule]
Average achievable rate [b/s/Hz]

Published with MATLAB® R2020b