

```

function [WeightMatrix_SNR, WeightMatrix_Rate] =
build_weight_matrices(Ps, channel_trace_map, elevation_quantized_all,
distance_ts_all, allowed_elevs, f_c, mode)

U = size(elevation_quantized_all, 3);
M = size(elevation_quantized_all, 1);
T = size(elevation_quantized_all, 2);

elevation_threshold = 25;
distance_m_all = distance_ts_all * 1e3;
GGU_dBi = 0; GSAT_dBi = 50;
GGU = 10^(GGU_dBi / 10); GSAT = 10^(GSAT_dBi / 10);
B = 5e6; N0_dBmHz = -174;
N0 = 10^((N0_dBmHz - 30)/10) * B;
c = 3e8;

samples_per_step = 1500; % corresponds to 10 sec step and 1.5 m/s
pedestrian

WeightMatrix_SNR = zeros(U, M, T);
WeightMatrix_Rate = zeros(U, M, T);

for k = 1:length(allowed_elevs)
    elev_val = allowed_elevs(k);

    for u = 1:U
        for m = 1:M
            elev_row = elevation_quantized_all(m, :, u);
            dist_row = distance_m_all(m, :, u);

            % Pre-index time steps where this elevation matches and is
            above threshold
            t_elev = find(elev_row == elev_val & elev_row >
elevation_threshold & ~isnan(dist_row));
            if isempty(t_elev), continue; end

            % Initialize fading trace and index
            h2_trace = channel_trace_map(elev_val);
            h2_trace_index = 1;

            for idx = 1:length(t_elev)
                t = t_elev(idx);
                d = dist_row(t);
                L = (c / (4 * pi * d * f_c))^2;
                rho = Ps * GGU * GSAT * L / N0;

                if strcmpi(mode, 'ideal')
                    if h2_trace_index > length(h2_trace)
                        [y_new, ~, ~, ~, ~, ~] =
coef_time_series_long_pedestrian(f_c, 'Suburban', elev_val, 5000);
                        h2_trace = abs(y_new).^2;
                        h2_trace_index = 1;
                    end
                    h_t = h2_trace(h2_trace_index);
                    h2_trace_index = h2_trace_index + samples_per_step;
                else
                    h_t = mean(h2_trace); % forecast mode uses average
power

```

```
end

    snr = rho * h_t;
    WeightMatrix_SNR(u, m, t) = 10 * log10(snr);
    WeightMatrix_Rate(u, m, t) = log2(1 + snr);
end
end
end
end
end
```