```matlab
function [Sh, Ih, Rh] = simulate_absir(M, Iv0, T, infection_rate, recovery_rate)
% Simulate agent-based transmission model. Uses a graph to represent social
% connectivity between agents.
%
% Note: You may find it helpful to summarize the state history matrices via
% summation. For instance sum(Ih, 1) will return the total number of
% infected persons at each timestep in the simulation.
%
% Inputs
%   M (square matrix): Adjacency matrix
%   Iv0 (column vector): Initial infection state
%   T (integer): Number of timesteps to simulate
%   infection_rate (float): Infection rate (probabilistic)
%   recovery_rate (float): Recovery rate (probabilistic)
%
% Returns
%   Sh (matrix): Susceptible state history
%   Ih (matrix): Infected state history
%   Rh (matrix): Recovered state history

    % Setup
    dim = length(Iv0);  % Dimensions of initial state
    Ih = zeros(dim, T); % Infection history
    Ih(:, 1) = Iv0; % Record the initial state to infection history
    Rh = zeros(dim, T); % Recovery history

    % Construct an "action" helper function
    function [I, R] = action(I, R)
        % Compute infection probabilities based on the social graph
        v_eff = infection_rate * M * I;
        v_pr = v_eff ./ (1 + v_eff);

        % Draw random values
        v_infect = rand(dim, 1) <= v_pr;
        v_recover = rand(dim, 1) <= recovery_rate;

        % Infect non-recovered individuals
        I = I | v_infect & (~R);
        % Recover infected individuals
        R = R | (I & v_recover);
        I = I & (~R);
    end

    % Run simulation
    for i = 2:T
        [Ih(:, i), Rh(:, i)] = action(Ih(:, i-1), Rh(:, i-1));
    end

    % Compute susceptible history
    Sh = ones(dim, T) - Ih - Rh;
end
```